TL
243
.B65

DOT HS-800 764

OPERATIONAL HYBRID COMPUTER SIMULATION FOR VEHICLE HANDLING STUDIES

DEPARTMENT OF
TRANSPORTATION

AUG 15 1974

LIBRARY

The Johns Hopkins University
Applied Physics Laboratory
8621 Georgia Avenue
Silver Spring, Maryland 20910

Contract No. DOT-HS-213-2-363
September 1972
Final Report

42

PREPARED FOR:
U.S. DEPARTMENT OF TRANSPORTATION
NATIONAL HIGHWAY TRAFFIC SAFETY ADMINISTRATION
WASHINGTON, D.C. 20590

The opinions, findings, and conclusions expressed in this publication are those of the authors and not necessarily those of the National Highway Traffic Safety Administration.

TL
243

TECHNICAL REPORT STANDARD TITLE PAGE

DEPARTMENT OF TRANSPORTATION
AUG 15 1974

1. Report No.
DOT/HS-800 764

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
OPERATIONAL HYBRID COMPUTER SIMULATION FOR VEHICLE HANDLING STUDIES

5. Report Date
September 1972

6. Performing Organization Code

7. Author(s)
P.F.Bohn, R.J.Keenan, J.Prowznik

8. Performing Organization Report No.
BCE-T-0367

9. Performing Organization Name and Address
The Johns Hopkins University
Applied Physics Laboratory
8621 Georgia Avenue
Silver Spring, Maryland 20910

10. Work Unit No.

11. Contract or Grant No.
DOT-HS-213-2-363

12. Sponsoring Agency Name and Address
U.S. Department of Transportation
National Highway Traffic Safety Admin.
Washington, D.C.

13. Type of Report and Period Covered
Final

14. Sponsoring Agency Code
NHTSA

15. Supplementary Notes

16. Abstract

A hybrid computer simulation for vehicle handling studies has been implemented, checked out, and validated. The simulation has been programmed to study both solid rear axle and independent rear suspension vehicles. The mathematical model used in the hybrid simulation was supplied by Bendix Research Laboratory.

Model validation was accomplished using parametric data for a 1971 Volkswagen Super Beetle. Braking, steering, and combinations of braking and steering were the inputs to the simulated mathematical model for the validation tests.

The hybrid computer simulation can be used for general studies of vehicle dynamics. Vehicles other than the Super Beetle can be simulated by using proper descriptive vehicle data.

17. Key Words
Hybrid Computer Simulation
Automobile Dynamics
Vehicle Mathematical Model
Vehicle Handling Study

18. Distribution Statement
unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
unclassified	unclassified	169	

Form DOT F 1700.7 (8-69)

ACKNOWLEDGMENT

The authors wish to acknowledge the assistance and cooperation they received from the Bendix Research Laboratory, Southfield, Michigan.

TABLE OF CONTENTS

Page

SECTION 1 - INTRODUCTION AND SUMMARY 1

SECTION 2 - FOUR-WHEELED VEHICLE HYBRID SIMULATION 3

2.1 Introduction 3
2.2 Simulation 3

2.2.1 Math Model 3
2.2.2 Allocation of Analog and Digital Computing Tasks 4
2.2.3 Implementation of Math Model 5

2.2.3.1 Analog Portion 5
2.2.3.2 Digital Portion 5

2.3 Validation 5

2.3.1 Selection of Validation Runs 6
2.3.2 Results of Validation Runs 6

2.3.2.1 Straight Line Braking 6
2.3.2.2 Trapezoidal Steer 6
2.3.2.3 Braking in a Turn 6

SECTION 3 - CONCLUSIONS AND RECOMMENDATIONS 13

APPENDIX A - FOUR-WHEELED VEHICLE MATH MODEL 15

1. Introduction 15
2. System Equations 15

2.1 Table of Contents 15
2.2 Equations of Motion (Ten Degrees of Freedom) 17
2.3 Vehicle Attitude and Position 19
2.4 Suspension Forces 19
2.5 Wheel Orientation 20
2.6 Resultant Forces and Moments 21
2.7 Radial Tire Force and Rolling Radius 23
2.8 Steering Equations 24
2.9 Wheel Rotational Equations 25
2.10 Wheel Slip 26
2.11 Tire Circumferential Force 26
2.12 Circumferential Friction Coefficient 26
2.13 Tire Side Force 26
2.14 Velocities of Tire Contact Points 29
2.15 Tire Moments 30
2.16 Wheel Camber with Respect to the Road 31
2.17 Brake and Drive Torques 31

3. Notation and List of Symbols 33

3.1 Notation 33
3.2 List of Symbols 33

Table of Contents (cont'd.) Page

APPENDIX B - FOUR-WHEELED VEHICLE HYBRID SIMULATION IMPLEMENTATION DOCUMENTATION 43

1. Listing of DSL/91 Digital Static Check Program 43
2. Listing of IBM 360/91 Digital Program 55
3. Analog Computer Diagrams 119
4. Wheel Springs and Shock Absorber Characteristics 127
5. Symbols and Definitions of Program Parameters 133

APPENDIX C - DESCRIPTION OF HYBRID COMPUTER SIMULATION LABORATORY 145

1. Hybrid Computer 145
2. Hybrid Software 148
3. Real-Time Hybrid 149
4. Hybrid Operation 150

REFERENCES 153

LIST OF ILLUSTRATIONS

Figure No.	Title	Page
2-1a	Time Histories - Straight Line Braking	8
2-1b	Time Histories - Straight Line Braking	9
2-2a	Time Histories - Trapezoidal Steer	10
2-2b	Time Histories - Trapezoidal Steer	11
2-3	Time Histories - Braking in a Turn	12
A-1	Hybrid Simulation Block Diagram of the Automobile	16
A-2	Rear Axle Representation	41
A-3	Analytical Representation of the Four-Wheeled Vehicle	42
B-1	A.C.D.* - Rotational Wheel Dynamics	121
B-2	A.C.D. - Continuous Resolution of Wheel Rates	122
B-3	A.C.D. - Suspension Forces and Deflections	123
B-4	A.C.D. - Wheel Springs and Shock Absorbers	124
B-5	A.C.D. - Radial Forces and Independent Rear Suspension	125
B-6	A.C.D. Steering System	126
B-7	Front Wheel Spring Force	129
B-8	Rear Wheel Spring Force	130
B-9	Front Shock Absorber Viscous Damping Force	131
B-10	Rear Shock Absorber Viscous Damping Force	132
C-1	Hybrid System Block Diagram	146

*Analog Computer Diagram

SECTION 1

INTRODUCTION AND SUMMARY

The Bendix Research Laboratories (BRL), Southfield, Michigan, has developed a Hybrid Computer Program of the Vehicle Handling Problem for NHTSA under Contracts No. FH-11-6946 and FH-11-7563. This program is used at the present time to investigate braking and component degradation. The mathematical model representing a complete operational vehicle system forms the basis of a large number of separate research efforts. NHTSA determined it necessary to increase the availability of the program to potential research institutions, who at the present time are not involved in investigations of vehicle handling behavior but who may get involved in such studies in the future (see Ref. 1). Therefore, it was decided that the Applied Physics Laboratory (APL) reprogram the BRL Vehicle Handling Simulation to operate on its hybrid computer.

In order for APL to reproduce the Vehicle Handling Simulation which existed at BRL, it was necessary to fully define the vehicle model. To accomplish this task, a meeting between APL and Bendix simulation personnel was arranged to discuss the status of the Bendix simulation, view the simulation in operation, discuss the analog and digital programs in detail, and obtain selected simulation data to enable APL to determine that it had successfully reproduced the simulation.

APL reprogrammed the Vehicle Simulation to operate on its hybrid computer. The program was checked against the existing one at BRL by utilizing an identical set of input data and comparing the output data with that of the Bendix simulation. Comparison was made using data representative of the split rear axle 1971 VW Super Beetle. No runs were attempted for a solid rear axle vehicle due to lack of data.

SECTION 2

FOUR-WHEELED VEHICLE HYBRID SIMULATION

2.1 INTRODUCTION

Contained in this section is a description of the four-wheeled vehicle hybrid computer simulation. The basic mathematical model, which was supplied by BRL (Refs. 2 and 10), is described in terms of its seventeen degrees of freedom. The perturbing forces and moments which act on the vehicle are also considered. The simulation implementation and checkout are discussed.

2.2 SIMULATION

2.2.1 Math Model

The seventeen-degrees-of-freedom vehicle model consists of:

1. A basic ten-degree-of-freedom model of the vehicle body, front wheels, and rear axle
2. A three-degree-of-freedom steering system model
3. A four-degree-of-freedom wheel rotational dynamics model

The basic ten-degree-of-freedom model regards the vehicle as an assembly of four rigid masses (the vehicle body, two front wheel masses, and the rear wheel axle combination). In the modified vehicle model, which includes independent rear suspension, each rear wheel is considered as an independent mass. The ten degrees of freedom consist of the six standard translational and rotational degrees of freedom for the body, two for the vertical motion of each front wheel, and two for the rotation and vertical motion of the rear axle.

The steering system model with three degrees of freedom represents the compliance in each of the front wheels and in the connecting rod. The tire moments about each king-pin axis are functions of the circumferential and side tire forces, the inclination and caster of the king pins, and the pneumatic trail effects of the tires. Steering wheel torque is the steering system input.

Four additional degrees of freedom (for a total of seventeen) are contained in the rotational equations of motion about the spin axis of each wheel. These equations, which include the differential effects of the rear wheels, yield the wheel rotation rates from which slip and, in turn, the variable friction coefficients are computed. The input to the equations can be either drive torque or brake torque.

The equations of motion of the vehicle body, wheels, and rear axle are perturbed by suspension, gravity, and tire forces and moments. The suspension equations include the effects of the springs, shock absorbers, and the auxiliary roll stiffness of the stabilizer bar.

The tire forces (radial, circumferential, and side) are computed for each wheel. The radial load is proportional to the distance between the wheel center and the road. The circumferential forces are a function of the radial load and circumferential coefficient of friction which, in turn, is a function of wheel slip. The side tire forces are a function of wheel slip, camber and wheel slip angle, and radial load.

2.2.2 Allocation of Analog and Digital Computing Tasks

The hybrid simulation block diagram of the automobile is shown in Figure A-1. The block diagram was derived from the digital and analog computer information supplied by BRL.

Calculated in the digital portion are the sprung mass equations of motion, wheel orientation angles, and tire force equations. Wheel brake and drive torques, velocities of the tire contact point, and resultant forces and moments are also computed in the digital portion.

The analog computations include the suspension forces, shock absorber and wheel spring functions, wheel slip, and circumferential coefficient of friction. In addition, the equations of motion of the unsprung masses and steering system equations are solved on the analog computer.

The hybrid simulation was time scaled to run at 1/10 real time, i.e., 10 seconds of machine time is equivalent to 1 second of real time.

2.2.3 Implementation of the Math Model

2.2.3.1 Analog Portion

The analog portion of the BRL hybrid vehicle simulation was programmed on an AD/4 analog computer manufactured by Applied Dynamics, Inc. The APL/JHU hybrid computer facility contains analog machines manufactured by Electronic Associates, Inc. Because of the many physical and electrical differences between the two analog machines, it was necessary to completely reprogram the analog portion of the BRL hybrid simulation. This reprogramming effort consisted of studying the BRL analog computer diagrams, drawing computer diagrams which were compatible with APL/JHU equipment, assigning analog computer equipment, patching the computer pre-patch panels and debugging the analog portion of the hybrid simulation.

To expedite the checkout and debug phases of the analog portion, a static analog test program was written. This was accomplished by using the math model equations in a digital simulation language program (Ref. 4). The digital program output provided an independent check of the simulation. The static check results verified that the programmed analog portion of the simulation represented the respective vehicle mathematical model equations.

2.2.3.2 Digital Portion

The digital portion of the BRL hybrid simulation was programmed on a Sigma V digital computer manufactured by the Xerox Data Systems (XDS). The APL/JHU hybrid facility has an IBM 360/91 digital computer (a complete description of the APL/JHU hybrid facility is given in Appendix C). Although both digital machines process Fortran IV programs, some program software changes were necessary. This was due, in part, to the differences which exist between the Sigma V and IBM 360/91 Fortran IV compilers.

2.3 VALIDATION

2.3.1 Selection of Validation Runs

Three types of vehicle maneuvers were selected to validate the simulation: namely, (1) straight line braking, (2) trapezoidal steer, and (3) braking in a turn. It was decided that since these runs involved the two primary simulation inputs and results of similar BRL runs had been published (Ref. 9), they would be sufficient to establish the operational status of the vehicle simulation.

2.3.2 Results of Validation Runs

2.3.2.1 Straight Line Braking

Figures 2-1a and 2-1b show the results of the straight-line braking tests. The initial and terminal velocities of the vehicle for the three tests were 30 and 10 mph, respectively. Three brake pedal pressures were used: 300, 600, and 900 psi. The time required to achieve full brake pressure for each test run was 1 second. No attempt was made to achieve wheel lockup. The time histories clearly show that the run time required to decelerate the vehicle from 30 to 10 mph is a linear function of applied brake force in the range $300 \leqq P_B \leqq 900$.

2.3.2.2 Trapezoidal Steer

The vehicle had an initial velocity of 30 mph and a zero steering command. As shown in Figure 2-2a, a positive trapezoidal steering command was input to the simulation. The steering command reached a maximum amplitude of 49 degrees in 0.4 second and was maintained for a duration of 3.5 seconds at which time the command was removed at a rate of 122 deg/sec. A second simulation run was made with the same conditions except that the initial steering command was negative (Figure 2-2b). These runs demonstrate the symmetrical response of the simulation to identical steering commands applied with a positive and negative sense.

2.3.2.3 Braking in a Turn

Starting with an initial vehicle velocity of

40 mph, the vehicle was placed in a 25 degree turn which was achieved in 0.5 seconds and maintained at this level for the duration of the run. All four wheels were braked 1.5 seconds after initiation of the steer command. Unequal brake pressures of 495 and 300 psi were applied to the front and rear wheels, respectively. Results of the test runs are exhibited in Figure 2-3.

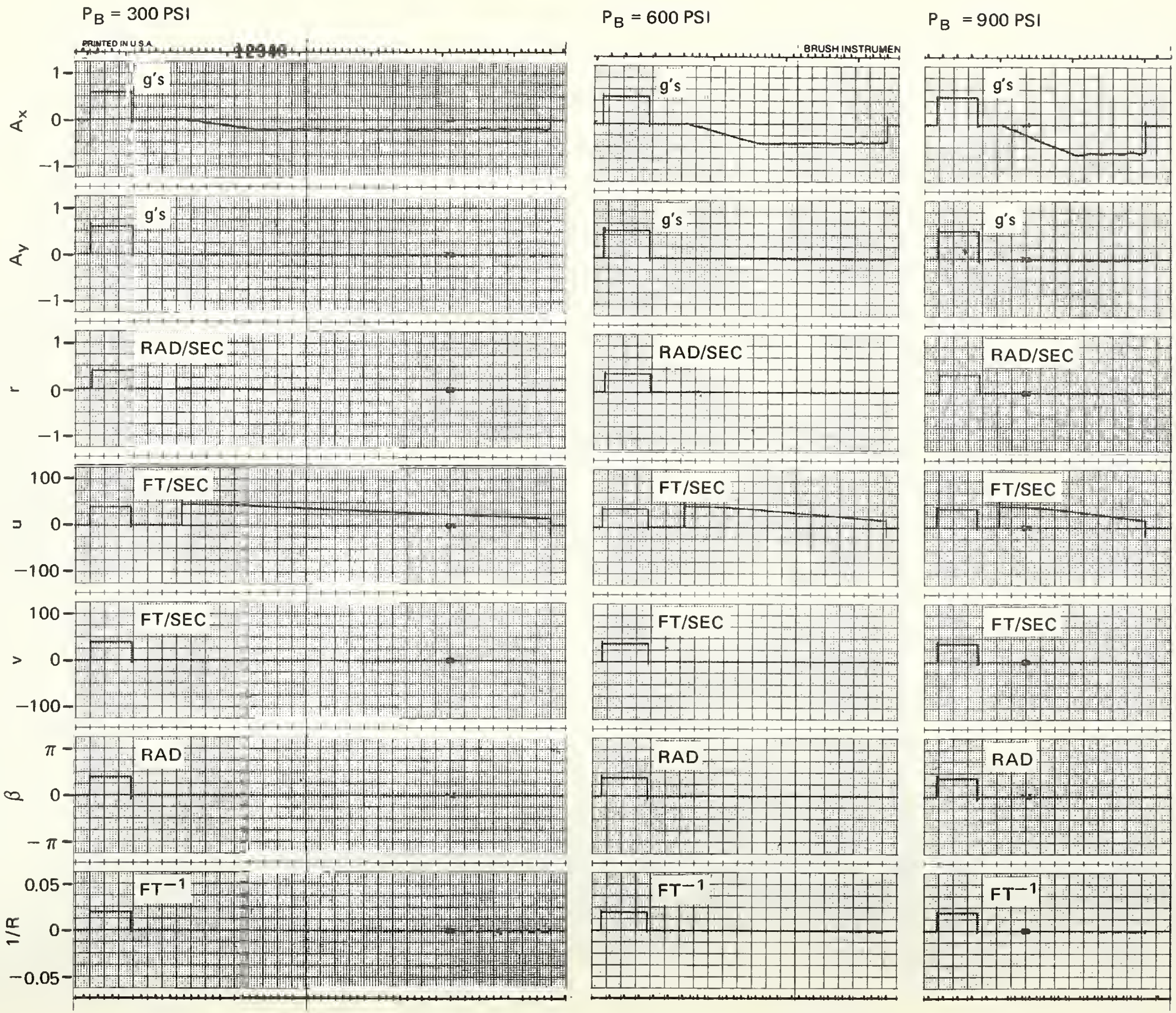


Fig. 2-1a TIME HISTORIES – STRAIGHT LINE BRAKING

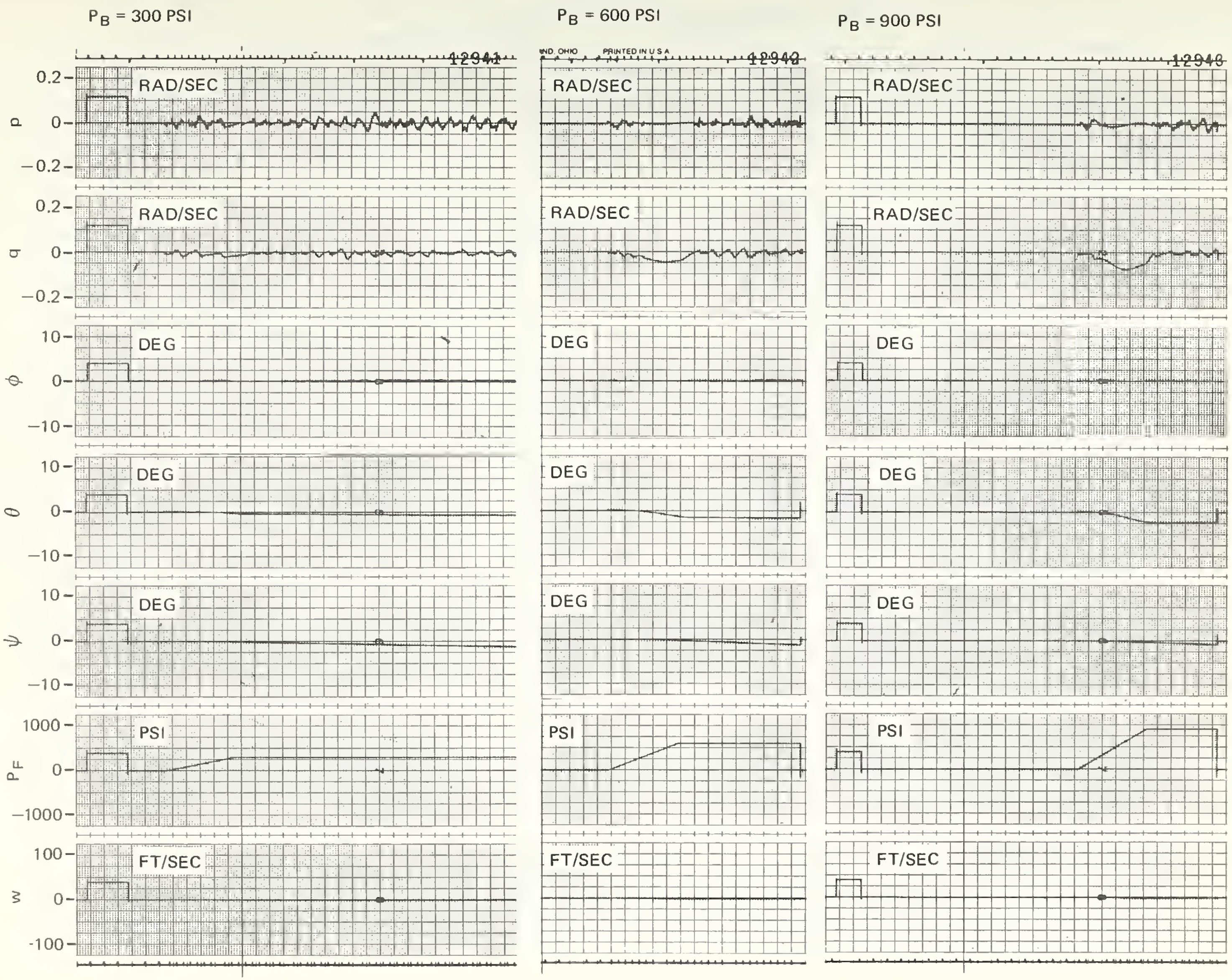


Fig. 2-1b TIME HISTORIES – STRAIGHT LINE BRAKING

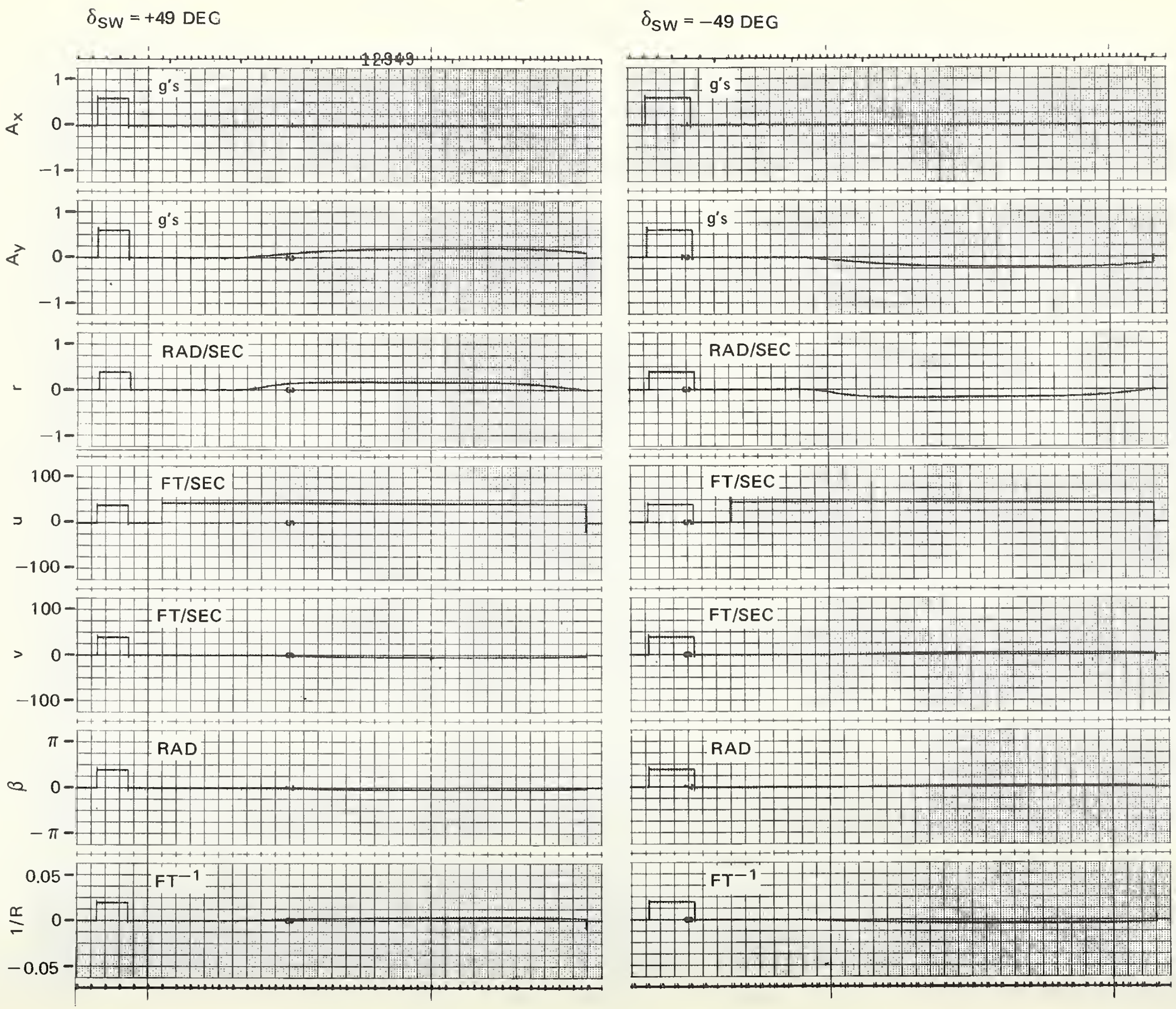


Fig. 2-2a TIME HISTORIES – TRAPEZOIDAL STEER

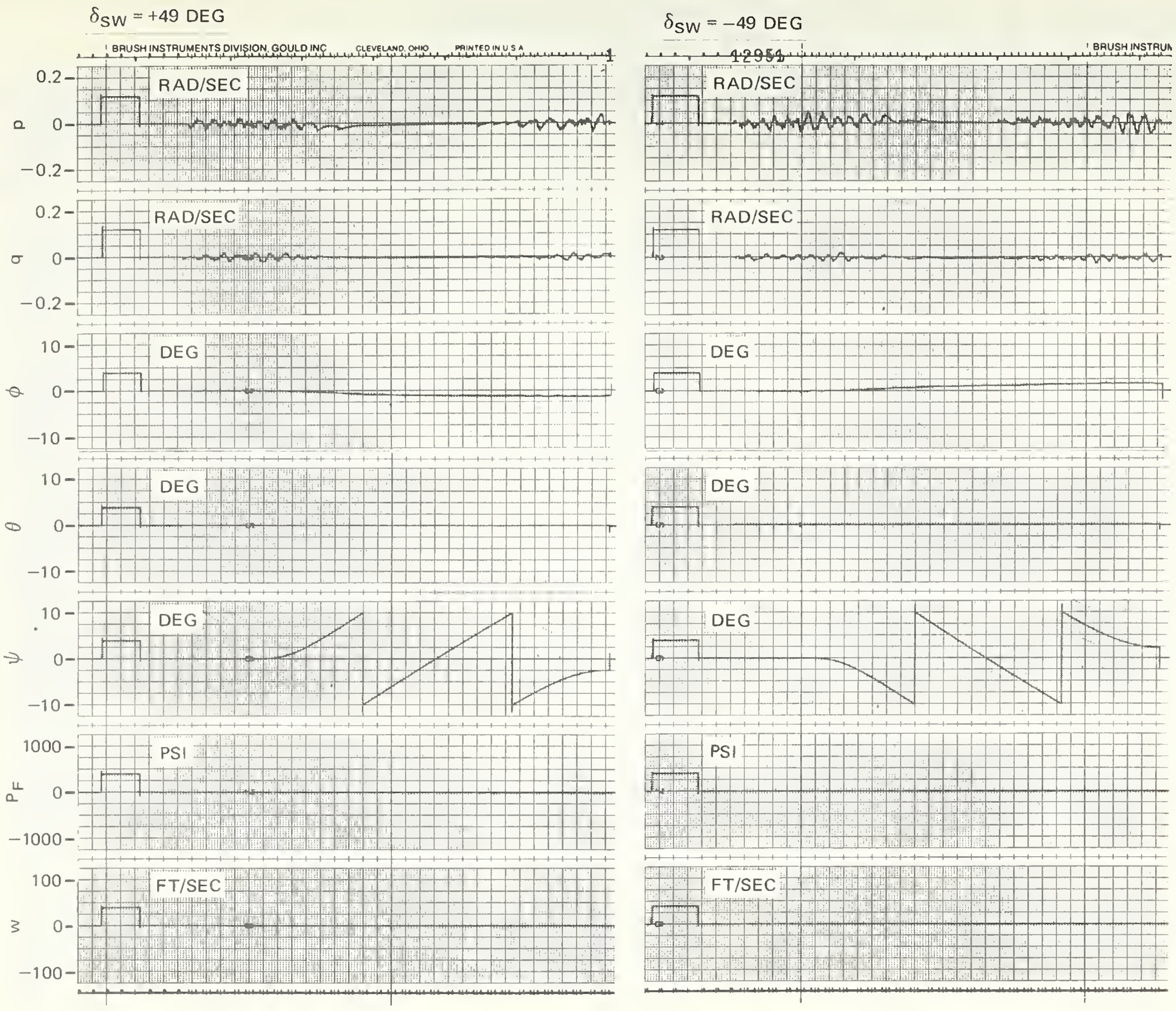


Fig. 2-2b TIME HISTORIES — TRAPEZOIDAL STEER

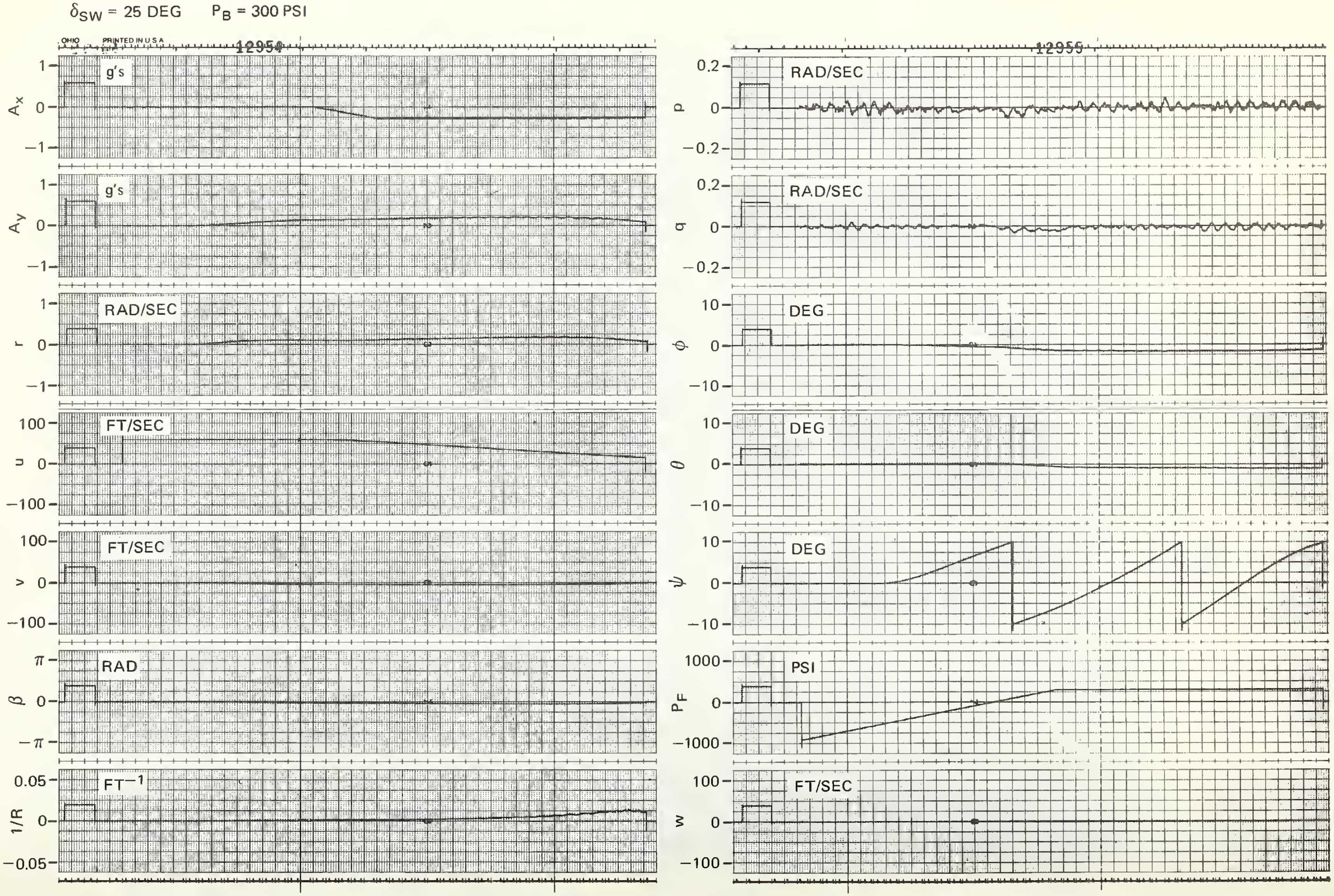


Fig. 2-3 TIME HISTORIES — BRAKING IN A TURN

SECTION 3

CONCLUSIONS AND RECOMMENDATIONS

This Vehicle Handling Simulation is operational and can be used for studying handling characteristics of four-wheel vehicles with independent rear suspension. The simulation has not been checked out and verified for the solid rear axle mode. Input data which defines the vehicle parameters must be supplied by the simulation user. Presently, APL has only a complete data deck for the 1971 VW Super Beetle.

It is recommended that the digital computer program be "cleaned-up" by discarding excess statements and subroutines. The excess statements arose from two sources. First, as the BRL digital program was converted to APL's digital computer, the original Bendix statements were commented and left in the program for reference purposes (see Appendix B). Second, the Bendix program contained old function generation code and subroutines which were no longer required. Removal of this code will decrease the size of the program and possibly decrease its execution time. It is also recommended that the model equations of Ref. 9 be compared with those existing in the APL program. Discrepancies should be explained and modifications made to the program if a modeling improvement would result.

APPENDIX A

FOUR-WHEELED VEHICLE MATH MODEL

1. INTRODUCTION

This appendix contains the vehicle mathematical model which was implemented on the APL/JHU hybrid computer. A hybrid simulation block diagram is shown in Figure A-1.

2. SYSTEM EQUATIONS

2.1 Table of Contents

Paragraph	Subject
2.2	Equations of Motion (10 Degrees of Freedom)
2.3	Vehicle Attitude and Position
2.4	Suspension Forces
2.5	Wheel Orientation
2.6	Resultant Forces and Moments
2.7	Radial Tire Force and Rolling Radius
2.8	Steering Equations
2.9	Wheel Rotational Equations
2.10	Wheel Slip
2.11	Tire Circumferential Force
2.12	Circumferential Friction Coefficient
2.13	Tire Side Force
2.14	Velocities of Tire Contact Points
2.15	Tire Moments
2.16	Wheel Camber with Respect to the Road
2.17	Brake and Drive Torques

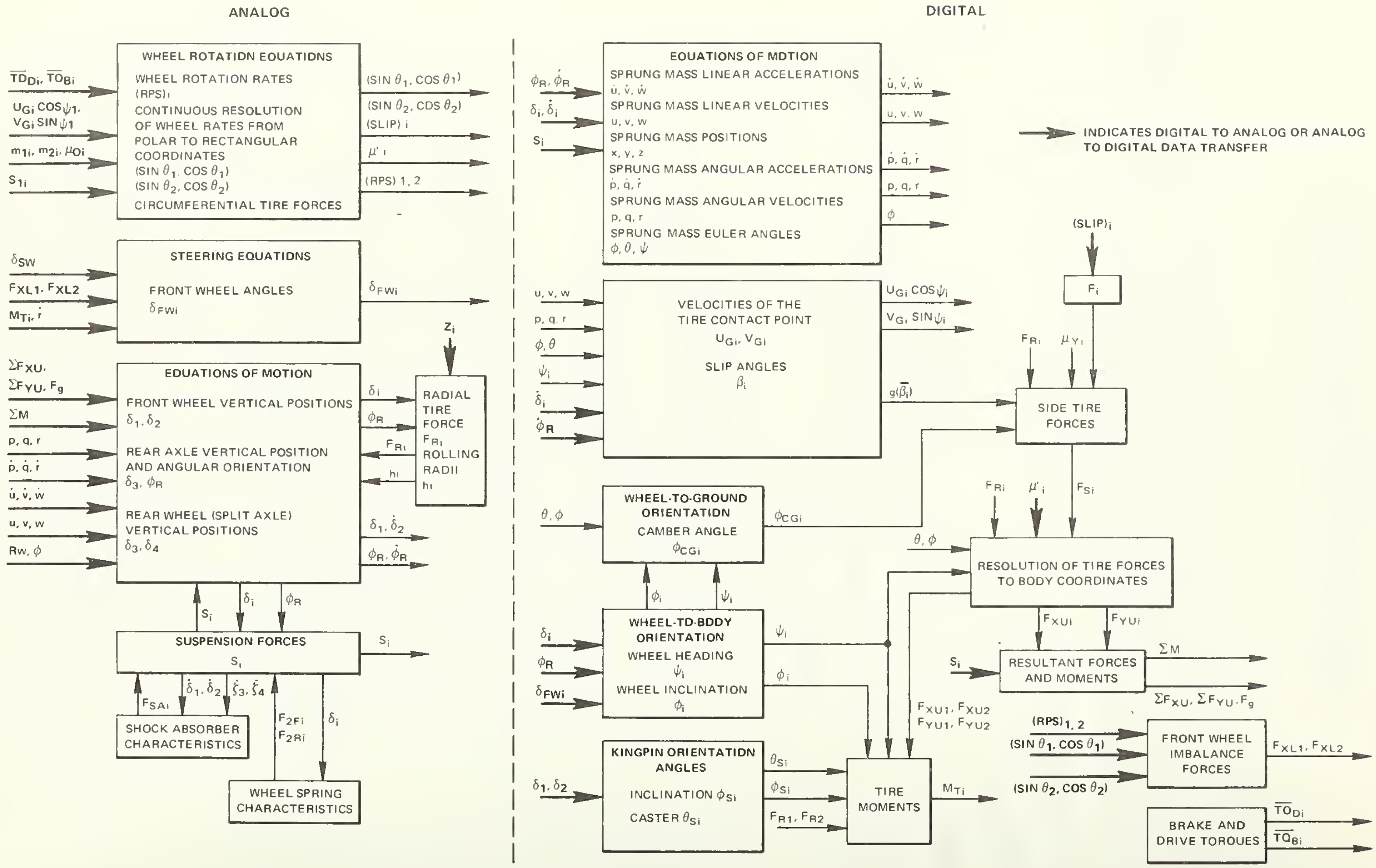

ANALOG
DIGITAL
INDICATES DIGITAL TO ANALOG OR ANALOG TO DIGITAL DATA TRANSFER
WHEEL ROTATION EQUATIONS
WHEEL ROTATION RATES (RPS)i
CONTINUOUS RESOLUTION OF WHEEL RATES FROM POLAR TO RECTANGULAR COORDINATES
(SIN θ_1, COS θ_1)
(SIN θ_2, COS θ_2)
CIRCUMFERENTIAL TIRE FORCES
$\overline{TD}_{Di}$, $\overline{TQ}_{Bi}$
U_{Gi} COSψ_1, V_{Gi} SINψ_1
m_{1i}, m_{2i}, μ_{Oi}
S_{1i}
(SIN θ_1, COS θ_1)
(SIN θ_2, COS θ_2)
(SLIP) i
μ'_i
(RPS) 1, 2
STEERING EQUATIONS
FRONT WHEEL ANGLES
δ_{FWi}
δ_{SW}
F_{XL1}, F_{XL2}
M_{Ti}, $\dot{r}$
δ_{FWi}
EQUATIONS OF MOTION
FRONT WHEEL VERTICAL POSITIONS
δ_1, δ_2
REAR AXLE VERTICAL POSITION AND ANGULAR ORIENTATION
δ_3, ϕ_R
REAR WHEEL (SPLIT AXLE) VERTICAL POSITIONS
δ_3, δ_4
ΣF_{XU}, ΣF_{YU}, F_g
ΣM
p, q, r
$\dot{p}$, $\dot{q}$, $\dot{r}$
$\dot{u}$, $\dot{v}$, $\dot{w}$
u, v, w
R_W, ϕ
δ_i
ϕ_R
F_{Ri}
h_i
$\dot{\delta}_1$, $\dot{\delta}_2$
ϕ_R, $\dot{\phi}_R$
Z_i
RADIAL TIRE FORCE
F_{Ri}
ROLLING RADII
h_i
S_i
δ_i
ϕ_R
SUSPENSION FORCES
S_i
S_i
F_{SAi}
$\dot{\delta}_1$, $\dot{\delta}_2$
$\dot{\zeta}_3$, $\dot{\zeta}_4$
F_{2Fi}
F_{2Ri}
δ_i
SHOCK ABSORBER CHARACTERISTICS
WHEEL SPRING CHARACTERISTICS
EQUATIONS OF MOTION
SPRUNG MASS LINEAR ACCELERATIONS
$\dot{u}$, $\dot{v}$, $\dot{w}$
SPRUNG MASS LINEAR VELOCITIES
u, v, w
SPRUNG MASS POSITIONS
x, y, z
SPRUNG MASS ANGULAR ACCELERATIONS
$\dot{p}$, $\dot{q}$, $\dot{r}$
SPRUNG MASS ANGULAR VELOCITIES
p, q, r
SPRUNG MASS EULER ANGLES
ϕ, θ, ψ
ϕ_R, $\dot{\phi}_R$
δ_i, $\dot{\delta}_i$
S_i
$\dot{u}$, $\dot{v}$, $\dot{w}$
u, v, w
$\dot{p}$, $\dot{q}$, $\dot{r}$
p, q, r
ϕ
VELOCITIES OF THE TIRE CONTACT POINT
U_{Gi}, V_{Gi}
SLIP ANGLES
β_i
u, v, w
p, q, r
ϕ, θ
$\dot{\psi}_i$
$\dot{\delta}_i$
$\dot{\phi}_R$
U_{Gi} COSψ_i
V_G, SINψ_i
$g(\overline{\beta}_i)$
(SLIP)$_i$
F_i
F_{Ri}
μ_{Yi}
SIDE TIRE FORCES
F_{Si}
F_{Ri}
μ'_i
WHEEL-TO-GROUND ORIENTATION
CAMBER ANGLE
ϕ_{CGi}
θ, ϕ
ϕ_{CGi}
ϕ_i
ψ_i
θ, ϕ
RESOLUTION OF TIRE FORCES TO BODY COORDINATES
F_{XUi}
F_{YUi}
WHEEL-TO-BODY ORIENTATION
WHEEL HEADING
ψ_i
WHEEL INCLINATION
ϕ_i
δ_i
ϕ_R
δ_{FWi}
ψ_i
ϕ_i
S_i
RESULTANT FORCES AND MOMENTS
ΣM
ΣF_{XU}, ΣF_{YU}, F_g
F_{XU1}, F_{XU2}
F_{YU1}, F_{YU2}
KINGPIN ORIENTATION ANGLES
INCLINATION ϕ_{Si}
CASTER θ_{Si}
δ_1, δ_2
θ_{Si}
ϕ_{Si}
F_{R1}, F_{R2}
TIRE MOMENTS
M_{Ti}
(RPS)$_{1, 2}$
(SIN θ_1, COS θ_1)
(SIN θ_2, COS θ_2)
FRONT WHEEL IMBALANCE FORCES
F_{XL1}, F_{XL2}
BRAKE AND DRIVE TORQUES
$\overline{TQ}_{Di}$
$\overline{TQ}_{Bi}$

Fig. A-1 HYBRID SIMULATION BLOCK DIAGRAM OF THE AUTOMOBILE

2.2 Equations of Motion (Ten Degrees of Freedom)

The equations of motion of the sprung mass, front wheel unsprung masses, and rear axle are presented below:

$$\Sigma M \dot{u} + \gamma_2 \dot{q} = \Sigma M (vr - wq - g \theta) + \Sigma F_{xu} \tag{1}$$

$$\Sigma M \dot{v} - \gamma_2 \dot{p} + \gamma_1 \dot{r} = \Sigma M (wp - ur + g \phi) + \Sigma F_{yu} \tag{2}$$

$$\begin{aligned} M_s \dot{w} = {} & M_s (uq - vp + g) - S_1 - S_2 - S_3 - S_4 + (F_{xu3} \\ & + F_{xu4}) T_{ANP} \end{aligned} \tag{3}$$

$$\begin{aligned} -\gamma_3 \dot{v} + (I_x + I_x') \dot{p} - (I_{xz} + I_{xz}') \dot{r} = {} & \gamma_3 (ur - wp - \\ & g \phi) + \Sigma N_{\phi u} \end{aligned} \tag{4}$$

$$\gamma_2 \dot{u} + (I_y + I_y') \dot{q} = \gamma_2 (vr - wq - g \theta) + \Sigma N_{\theta u} \tag{5}$$

$$\begin{aligned} \gamma_1 \dot{v} - (I_{xz} + I_{xz}') \dot{p} + (I_z + I_R + I_z') \dot{r} = {} & \gamma_1 \\ & (wp - ur + g \phi) + \Sigma N_{\Psi u} \end{aligned} \tag{6}$$

$$\begin{aligned} \frac{M_{uF}}{2} \dot{w} + \frac{M_{uF} T_F}{4} \dot{p} - \frac{M_{uF}\, a}{2} \dot{q} + \frac{M_{uF}}{2} \ddot{\delta}_1 = {} & \frac{M_{uF}}{2} (uq - vp + \\ & g) + F_{zu1} + S_1 - F_{yu1} \text{ TAN} \left(\frac{2H_{FC}}{T_F} \right) \end{aligned} \tag{7}$$

$$\begin{aligned} \frac{M_{uF}}{2} \dot{w} - \frac{M_{uF} T_F}{4} \dot{p} - \frac{M_{uF}\, a}{2} \dot{q} + \frac{M_{uF}}{2} \ddot{\delta}_2 = {} & \frac{M_{uF}}{2} (uq - vp + \\ & g) + F_{zu2} + S_2 + F_{yu2} \text{ TAN} \left(\frac{2H_{FC}}{T_F} \right) \end{aligned} \tag{8}$$

$$\begin{aligned} M_{uR} \dot{w} + b\, M_{uR} \dot{q} + \ddot{\delta}_3 M_{uR} = {} & M_{uR} (uq - vp + g) + \\ & F_{zu3} + F_{zu4} + S_3 + S_4 + (F_{xu3} + F_{xu4}) T_{ANP} \end{aligned} \tag*{(9)*}$$

*Solid Rear Axle

$$I_R \dot{p} + I_R \ddot{\phi}_R = \Sigma N_{\phi R} - (F_{yu3} + F_{yu4}) (C_{RRC} \delta_3 + H_{RC}) + \left(\frac{T_S}{2}\right)\left(F_{xu3} - F_{xu4}\right) T_{ANP} \qquad (10)*$$

$$\frac{M_{uR}}{2} \dot{w} + \frac{M_{uR} b\dot{q}}{2} + \frac{M_{uR} T_R \dot{p}}{4} + \ddot{\delta}_3 \frac{M_{uR}}{2} = \frac{M_{uR}}{2} (uq - vp + g) + F_{zu3} + S_3 - F_{yu3} \mathrm{TAN}\left(\frac{2H_{RC}}{T_R}\right) \qquad (11)**$$

$$\frac{M_{uR}}{2} \dot{w} + \frac{M_{uR}}{2} b\dot{q} - \frac{M_{uR} T_R \dot{p}}{4} + \ddot{\delta}_4 \frac{M_{uR}}{2} = \frac{M_{uR}}{2} (uq - vp + g) + F_{zu4} + S_4 + F_{yu4} \mathrm{TAN}\left(\frac{2H_{RC}}{T_R}\right) \qquad (12)**$$

where

$$\Sigma M = M_s + M_{uF} + M_{uR} \qquad (13)$$

$$I'_x = M_{uF} z_F^2 + M_{uR} z_R^2 \qquad (14)$$

$$I'_y = I'_x \qquad (15)$$

$$I'_z = M_{uF} (a^2 + \frac{{T_F}^2}{4}) + M_{uR} b^2 \qquad (16)*$$

$$I'_z = M_{uF} \left(a^2 + \frac{{T_F}^2}{4}\right) + M_{uR} \left(b^2 + \frac{{T_R}^2}{4}\right) \qquad (17)**$$

$$I'_{xz} = M_{uF} a z_F - M_{uR} b z_R \qquad (18)$$

$$\gamma_1 = M_{uF} a - M_{uR} b \qquad (19)$$

$$\gamma_2 = M_{uF} z_F + M_{uR} z_R \qquad (20)$$

$$\gamma_3 = \gamma_2 \qquad (21)$$

*Solid Rear Axle

**Split Rear Axle

2.3 Vehicle Attitude and Position

The Euler angles and x, y, z coordinates in fixed space of the sprung mass are computed by the following equations:

$$\phi = \int_o^t (p + r\theta)\, dt + \phi(o) \qquad (22)$$

$$\theta = \int_o^t (q - r\phi)\, dt + \theta(o) \qquad (23)$$

$$\psi = \int_o^t (r + q\, \phi)\, dt + \psi(o) \qquad (24)$$

$$x = \int_o^t (u \text{ COS } \psi - v \text{ SIN } \psi)\, dt + x(o) \qquad (25)$$

$$y = \int_o^t (u \text{ SIN } \psi + v \text{ COS } \psi)\, dt + y(o) \qquad (26)$$

$$z = \int_o^t (-u\theta + v\phi + w)\, dt + z(o) \qquad (27)$$

2.4 Suspension Forces

Suspension forces, including effects of viscous and coulomb damping, suspension stops and auxiliary roll stiffness for front and rear are presented below:

For i = 1, 2 (Front)

$$S_i = \frac{b}{2(a + b)} M_s g - F_{SAi} - F_{1Fi} - F_{2Fi} - (-1)^i \frac{R_F (\delta_2 - \delta_1)}{T_F^2} \qquad (28)$$

where

$$F_{1Fi} = C'_f \text{ sgn } \dot{\delta}_i \qquad (29)$$

$$\begin{aligned} F_{2Fi} &= K_F\, \delta_i, && \text{for } \Omega_{FC} \leq \delta_i \leq \Omega_{FT} \qquad (30) \\ &= K_F \left[\lambda_F\, \delta_i - (\lambda_F - 1)\, \Omega_{FC}\right], && \text{for } \delta_i < \Omega_{FC} \\ &= K_F \left[\lambda_F\, \delta_i - (\lambda_F - 1)\, \Omega_{FT}\right], && \text{for } \delta_i > \Omega_{FT} \end{aligned}$$

$$S_i = \frac{a}{2(a + b)} M_s g - F_{SAi} - F_{1Ri} - F_{2Ri} + (-1)^i \left[\frac{R_R \phi_R}{T_s}\right] \quad (31)^*$$

$$S_i = \frac{a}{2(a+b)} M_s g - F_{SAi} - F_{1Ri} - F_{2Ri} + (-1)^{i-1} \left[\frac{R_R}{T_R^{\,2}}\right]\left(\delta_4 - \delta_3\right) \quad (31\text{-A})^{**}$$

$$F_{1Ri} = C'_R \operatorname{sgn} \dot{\zeta}_i \quad (32)$$

For i = 3,4 (Rear)

$$\begin{aligned} F_{2Ri} &= K_R \zeta_i, && \text{for } \Omega_{RC} \leqq \zeta_i \leqq \Omega_{RT} \quad (33) \\ &= K_R \left[\lambda_R \zeta_i - (\lambda_R - 1) \Omega_{RC}\right], && \text{for } \zeta_i < \Omega_{RC} \\ &= K_R \left[\lambda_R \zeta_i - (\lambda_R - 1) \Omega_{RT}\right], && \text{for } \zeta_i > \Omega_{RT} \end{aligned}$$

and

$$\zeta_3 = \frac{T_s}{2} \phi_R + \delta_3 \quad (34)^*$$

$$\dot{\zeta}_3 = \frac{T_s}{2} \dot{\phi}_R + \dot{\delta}_3 \quad (35)^*$$

$$\zeta_4 = - \frac{T_s}{2} \phi_R + \delta_3 \quad (36)^*$$

$$\dot{\zeta}_4 = - \frac{T_s}{2} \dot{\phi}_R + \dot{\delta}_3 \quad (37)^*$$

$$\zeta_3 = \delta_3 \quad (38)^{**}$$

$$\dot{\zeta}_3 = \dot{\delta}_3 \quad (39)^{**}$$

$$\zeta_4 = \delta_4 \quad (40)^{**}$$

$$\dot{\zeta}_4 = \dot{\delta}_4 \quad (41)^{**}$$

2.5 Wheel Orientation

The orientations of the wheels with respect to the sprung mass are defined by the following equations:

$$\phi_3 = \phi_4 = \phi_R \quad (42)^*$$

---

*Solid Rear Axle

**Split Rear Axle

$$\phi_1 = f(\delta_1, \phi_{FNT}) + F_{EE1} \operatorname{sgn} F_{S1} \quad (43)$$

$$\phi_2 = - f(\delta_2, \phi_{FNT}) + F_{EE2} \operatorname{sgn} F_{S2} \quad (44)$$

$$\phi_3 = f(\delta_3, \phi_{RR}) \quad (45)^{**}$$

$$\phi_4 = f(\delta_4, \phi_{RR}) \quad (46)^{**}$$

$$\Psi_1 = \delta_{FW1} + f(\delta_1, \Psi_{FNT}) + \varepsilon_{K1} \quad (47)$$

$$\Psi_2 = \delta_{FW2} - f(\delta_2, \Psi_{FNT}) + \varepsilon_{K2} \quad (48)$$

$$\Psi_3 = K_{RS} \phi_R \quad (49)^{*}$$

$$\Psi_4 = \Psi_3 \quad (50)^{*}$$

$$\Psi_3 = f(\delta_3, \Psi_{RR}) \quad (51)^{**}$$

$$\Psi_4 = f(\delta_4, \Psi_{RR}) \quad (52)^{**}$$

2.6 Resultant Forces and Moments

The resultant tire and suspension forces and moments, including the anti-pitch and wheel imbalance, required for the equations of motion are given below:

$$\Sigma F_{xu} = \sum_{i=1}^{4} F_{xui} \quad (53)$$

$$F_{xui} = F_{Ri} \theta + F_{Ci} \cos \Psi_i - F_{Si} \sin \Psi_i \quad (54)$$

$$\Sigma F_{yu} = \sum_{i=1}^{4} F_{yui} \quad (55)$$

$$F_{yui} = -F_{Ri} \phi + F_{Ci} \sin \Psi_i + F_{Si} \cos \Psi_i \quad (56)$$

*Solid Rear Axle

**Split Rear Axle

$$F_{zui} = -F_{Ri} \tag{57}$$

$$F_{RZ1} = -F_{xu1}\left[\frac{Z_{RO} - Z_R - (Z_F + \delta_1 + h_1)}{L_F}\right] \tag{58}$$

$$F_{RZ2} = -F_{xu2}\left[\frac{Z_{RO} - Z_R - (Z_F + \delta_2 + h_2)}{L_F}\right] \tag{59}$$

$$F_{XL1} = M_{L1}\ R_{RIM}\ \mathrm{RPS1}\ \mathrm{RPS1}\ \mathrm{COS}\ \theta_{L1} \tag{60}$$

$$F_{XL2} = M_{L2}\ R_{RIM}\ \mathrm{RPS2}\ \mathrm{RPS2}\ \mathrm{COS}\ \theta_{L2} \tag{61}$$

$$F_{ZL1} = M_{L1}\ R_{RIM}\ \mathrm{RPS1}\ \mathrm{RPS1}\ \mathrm{SIN}\ \theta_{L1} \tag{62}$$

$$F_{ZL2} = M_{L2}\ R_{RIM}\ \mathrm{RPS2}\ \mathrm{RPS2}\ \mathrm{SIN}\ \theta_{L2} \tag{63}$$

$$\begin{aligned} \Sigma N_{\psi u} = {} & (F_{yu1} + F_{yu2})\ a - (F_{yu3} + F_{yu4})\ b + \\ & (F_{xu2} - F_{xu1})\ \frac{T_F}{2} + (F_{xu4} - F_{xu3})\ \frac{T_R}{2} \end{aligned} \tag{64}$$

$$\begin{aligned} \Sigma N_{\phi u} = {} & \frac{T_F}{2}\ (S_2 - S_1) + \frac{T_S}{2}\ (S_4 - S_3) - \\ & F_{yu1}\ (Z_F + \delta_1 + h_1 - H_{FC}) - \\ & F_{yu2}\ (Z_F + \delta_2 + h_2 - H_{FC}) - \\ & \left(\frac{T_{LF}}{2}\right)\left(F_{RZ1} - F_{RZ2}\right) - \left(\frac{T_S}{2}\right)\left(F_{xu3} - F_{xu4}\right)\ T_{ANP} - \\ & (F_{yu3} + F_{yu4})\ (C_{RRC}\ \delta_3 + H_{RC} + Z_R) \end{aligned} \tag{65*}$$

$$\begin{aligned} \Sigma N_{\theta u} = {} & (S_1 + S_2)\ a - (S_3 + S_4)\ b + F_{xu1}\ (z_F + \delta_1 + h_1) \\ & + F_{xu2}\ (z_F + \delta_2 + h_2) + F_{xu3}\left(z_R + \delta_3 + \frac{T_R}{2}\ \phi_R + h_3\right) \\ & + F_{xu4}\left(z_R + \delta_3 - \frac{T_R}{2}\ \phi_R + h_4\right) - b\ (F_{xu3} + F_{xu4})\ T_{ANP} \\ & + a\ (F_{RZ1} + F_{RZ2}) \end{aligned} \tag{66*}$$

*Solid Rear Axle

$$\Sigma N_{\phi u} = \frac{T_F}{2} (S_2 - S_1) + \frac{T_R}{2} (S_4 - S_3) \tag{67**}$$

$$- F_{yu1} (Z_F + \delta_1 + h_1 - H_{FC})$$

$$- F_{yu2} (Z_F + \delta_2 + h_2 - H_{FC})$$

$$- F_{yu3} (Z_R + \delta_3 + h_3 - H_{RC}) - F_{yu4} (Z_R + \delta_4 + h_4 - H_{RC})$$

$$\Sigma N_{\theta u} = (S_1 + S_2)a - (S_3 + S_4)\ b \tag{68**}$$

$$+ F_{xu1} (Z_F + \delta_1 + h_1) + F_{xu2} (Z_F + \delta_2 + h_2)$$

$$+ F_{xu3} (Z_R + \delta_3 + h_3) + F_{xu4} (Z_R + \delta_4 + h_4)$$

$$\Sigma N_{\phi R} = F_{zu3}\left(\frac{T_R}{2} + h_3\phi\right) - F_{zu4}\left(\frac{T_R}{2} - h_4\phi\right) \tag{69}$$

$$- F_{yu3}\left(- \frac{T_R}{2}\phi + h_3\right) - F_{yu4}\left(\frac{T_R}{2}\phi + h_4\right)$$

$$+ (S_3 - S_4) \frac{T_S}{2}$$

2.7 Radial Tire Force and Rolling Radius

The radial tire forces and the rolling radii of the tires are computed by the following equations:

$$F_{Ri} = K_{Ti} (R_w - h_i) \tag{70}$$

where the initial tire loading and orientation are shown below:

$$\theta_{(0)} = \frac{\left[h_{1(0)} - h_{3(0)}\right] + \left[Z_F - Z_R\right]}{a+b}$$

$$h_{1(0)} = h_{2(0)} = R_w - \frac{g}{2K_{T1}} \left[M_{UF} + \left(\frac{b}{a+b}\right) M_S\right]$$

$$h_{3(0)} = h_{4(0)} = R_w - \frac{g}{2K_{T3}} \left[M_{UR} + \left(\frac{a}{a+b}\right) M_S\right]$$

$$Z_{(0)} = \frac{b\left[h_{1(0)} + Z_F\right] + a\left[h_{3(0)} + Z_R\right]}{a+b}$$

**Split Rear Axle

where

$$h_i = - Z_i \; ; \; i = 1,2,3,4$$

$$Z_1 = Z - a\,\theta + \frac{T_F}{2}\,\phi + Z_F + \delta_1 \qquad (71)$$

$$Z_2 = Z - a\,\theta - \frac{T_F}{2}\,\phi + Z_F + \delta_2 \qquad (72)$$

$$Z_3 = Z + b\,\theta + \frac{T_R}{2}\,\phi + Z_R + \frac{T_R}{2}\,\phi_R + \delta_3 \qquad (73)^*$$

$$Z_4 = Z + b\,\theta - \frac{T_R}{2}\,\phi + Z_R - \frac{T_R}{2}\,\phi_R + \delta_3 \qquad (74)^*$$

$$Z_3 = Z + b\,\theta + \frac{T_R}{2}\,\phi + Z_R + \delta_3 \qquad (75)^{**}$$

$$Z_4 = Z + b\,\theta - \frac{T_R}{2}\,\phi + Z_R + \delta_4 \qquad (76)^{**}$$

2.8 Steering Equations

The steering equations are presented below:

$$(\dot{r} + \ddot{\delta}_{FWi})\, I_{FW} = -H_i\, \dot{\delta}_{FWi} + M_{Ti} - M_{SSi} \qquad (77)$$

$$i = 1,2$$

$$M_{CR}\, \ddot{y}_{CR} = -C_{FCR} - C_{CR}\, \dot{y}_{CR} + \frac{T_p}{a_p} + \frac{M_{SS1}}{a_1} + \frac{M_{SS2}}{a_2} \qquad (78)$$

where $C_{FCR} = f(\dot{y}_{CR})$

conditions:

$$T_p = N_G \left\{ K_{SC} \left(\delta_{SW} - N_G\, \delta_p - \frac{\epsilon_{SP}}{2} \right) \right\} \; ; \quad \left| \delta_{SW} - N_G\, \delta_p \right| > \frac{\epsilon_{SP}}{2}$$

otherwise $T_p = 0$

$$M_{SSi} = K_{SLi} \left[\left(\delta_{FWi} - \frac{y_{CR}}{a_i} \right) - \frac{\epsilon_{pi}}{2} \right] ; \quad \left| \delta_{FWi} - \frac{y_{CR}}{a_i} \right| > \frac{\epsilon_{pi}}{2}$$

**Split Rear Axle
*Solid Rear Axle

otherwise $M_{SSi} = 0$

2.9 Wheel Rotational Equations

The wheel rotational equations required to compute wheel slip are presented below:

$$I_{WF} \frac{d}{dt} (RPS)_1 = -F_{C1} h_1 + \overline{TQ}_1 \lambda_{B1} \tag{79}$$

$$I_{WF} \frac{d}{dt} (RPS)_2 = -F_{C2} h_2 + \overline{TQ}_2 \lambda_{B2} \tag{80}$$

$$\left[I_{WR} + \frac{I_D (\overline{AR})^2}{4}\right] \left[\frac{d}{dt} (RPS)_3\right] + \left[\frac{I_D (\overline{AR})^2}{4}\right] \left[\frac{d}{dt} (RPS)_4\right] = -F_{C3} h_3 + \overline{TQ}_3 \lambda_{B3} \tag{81}$$

$$\left[\frac{I_D (\overline{AR})^2}{4}\right] \left[\frac{d}{dt} (RPS)_3\right] + \left[I_{WR} + \frac{I_D (\overline{AR})^2}{4}\right] \left[\frac{d}{dt} (RPS)_4\right] = -F_{C4} h_4 + \overline{TQ}_4 \lambda_{B4} \tag{82}$$

where

$$(RPS)_i = (RPS)_{io} + \int_o^t \frac{d}{dt} \left[(RPS)_i\right] dt \tag{83}$$

For $(SLIP)_i = 0$ at $t = 0$

$$(RPS)_{io} = \frac{u_{Gi}(o) \cos \Psi_i(o) + v_{Gi}(o) \sin \Psi_i(o)}{h_i(o)} \tag{84}$$

and where

$$\overline{TQ}_1 = \overline{TQ}_{B1} \tag{85}$$

$$\overline{TQ}_2 = \overline{TQ}_{B2} \tag{86}$$

$$\overline{TQ}_3 = \frac{\overline{AR}}{2} \overline{TQ}_D + \overline{TQ}_{B3} \tag{87}$$

$$\overline{TQ}_4 = \frac{\overline{AR}}{2} \overline{TQ}_D + \overline{TQ}_{B4} \tag{88}$$

2.10 Wheel Slip

Computation of wheel slip is performed by the following equations:

$$(\mathrm{SLIP})_i = 1 \quad \text{for } \xi_i > 1 \tag{89}$$

$$= \xi_i \quad \text{for } -1 \leqq \xi_i \leqq 1$$

$$= -1 \quad \text{for } \xi_i < -1$$

where

$$\xi_i = 1 - \frac{(\mathrm{RPS})_i \, h_i}{u_{Gi} \, \mathrm{COS} \, \Psi_i + v_{Gi} \, \mathrm{SIN} \, \Psi_i} \tag{90}$$

2.11 Tire Circumferential Force

The circumferential tire forces for both driving and braking are defined below:

$$F_{Ci} = - \mu_i' \, F_{Ri} \tag{91}$$

2.12 Circumferential Friction Coefficient

The circumferential friction coefficient equations are shown below:

$$\mu_i' = m_{2i} \, (\mathrm{SLIP})_i + \mu_{0i} \quad \text{for } (\mathrm{SLIP})_i > S_{Ii} \tag{92}$$

$$= m_{1i} \, (\mathrm{SLIP})_i \quad \text{for } (\mathrm{SLIP})_i \leqq S_{Ii}$$

where

$$m_{2i} = \mu_{1i} - \mu_{0i} \tag{93}$$

$$S_{Ii} = \frac{\mu_{0i}}{m_{1i} - m_{2i}} \tag{94}$$

2.13 Tire Side Force

The nonlinear tire side forces are computed using the following equations:

Preliminary computations:

$$\beta_i = \tan^{-1} \left[\frac{v_{Gi}}{|u_{Gi}|} \right] - \Psi_i \, \mathrm{sgn} \, u_{Gi} \tag{95}$$

If $F_{Ri} \leqq \Omega_T A_2$,

then let

$$\beta_i' = \frac{A_2 A_3 (A_4 - F_{Ri}) F_{Ri} \emptyset_{CGi}}{A_4 \left[A_1 F_{Ri} (F_{Ri} - A_2) - A_o A_2\right]} \qquad i = 1,2 \qquad (96)$$

$$\bar{\beta}_i = \frac{A_1 F_{Ri} (F_{Ri} - A_2) - A_o A_2}{A_2 \mu_{yi} F_{Ri}} (\beta_i + \beta_i') \qquad (97)$$

If $F_{Ri} \leqq \Omega_T RA_2$,

then let $\qquad i = 3,4$

$$\beta_i' = \frac{RA_o RA_3 (RA_4 - F_{Ri}) F_{Ri} \emptyset_{CGi}}{RA_4 \left[RA_1 F_{Ri} (F_{Ri} - RA_2) - RA_o RA_2\right]} \qquad (96\text{-A})$$

$$\bar{\beta}_i = \frac{RA_1 F_{Ri} (F_{Ri} - RA_2) - RA_o RA_2}{RA_2 \mu_{yi} F_{Ri}} (\beta_i + \beta_i') \qquad (97\text{-A})$$

The side coefficient of friction is defined below:

$$\mu_{yi} = (0.85 + .0055\ C_v)\ SN\ (.025) \qquad i = 1,2,3,4 \qquad (97\text{-B})$$

and

$$C_{v_i} = \sqrt{U^2 + V^2} \qquad (97\text{-C})$$

If $F_{Ri} > \Omega_T A_2$,

then let

$$\beta_i' = \frac{A_2 A_3 \Omega_T (A_4 - \Omega_T A_2) \emptyset_{CGi}}{A_4 \left[A_1 A_2 \Omega_T (\Omega_T - 1) - A_o\right]} \qquad i = 1,2 \qquad (98)$$

$$\bar{\beta}_i = \frac{A_1 A_2 \Omega_T (\Omega_T - 1) - A_o}{\mu_{yi} F_{Ri}} (\beta_i + \beta_i') \qquad (99)$$

If $F_{Ri} > \Omega_T \; RA_2$,

then let

$$\beta'_i = \frac{RA_2 \; RA_3 \; R\Omega_T \; (RA_4 - R\Omega_T \; RA_2) \; \phi_{CGi}}{RA_4 \left[RA_1 \; RA_2 \; R\Omega_T \; (R\Omega_T - 1) - RA_o \right]} \qquad i = 3,4 \tag{98-A}$$

$$\bar{\beta}_i = \frac{RA_1 \; RA_2 \; R\Omega_T \; (R\Omega_T - 1) - RA_o}{\mu_{yi} \; F_{Ri}} \; (\beta_i + \beta'_i) \tag{99-A}$$

Define

$$g(\bar{\beta}_i) = \bar{\beta}_i - \frac{1}{3} \bar{\beta}_i \left| \bar{\beta}_i \right| + \frac{1}{27} \bar{\beta}_i^{\,3} \text{ if } \left| \bar{\beta}_i \right| < 3 \tag{100}$$

$$= \frac{\bar{\beta}_i}{\left| \bar{\beta}_i \right|} \quad \text{if } \left| \bar{\beta}_i \right| \geqq 3 \qquad i = 1,2,3,4$$

Side force computations:

$$F_{Si} = \mu_{yi} \; F_{Ri} \; g\,(\bar{\beta}_i) \; \left[F_i \right] \qquad i = 1,2,3,4 \tag{101}$$

where

$$\text{sgn } F_{Si} = \text{sgn } \bar{\beta}_i \tag{102}$$

and where

$$F_i = f(\text{SLIP})_i \tag{103}$$

The quantity μ_{yi} is empirically derived from test data.

2.14 Velocities of Wheel Centers

The velocities of the wheel centers along the vehicle axes are computed by the following equations:

$$u_1 = u - \frac{T_F}{2} r + z_F q \qquad (104)$$

$$u_2 = u + \frac{T_F}{2} r + z_F q \qquad (105)$$

$$u_3 = u - \frac{T_R}{2} r + z_R q \qquad (106)$$

$$u_4 = u + \frac{T_R}{2} r + z_R q \qquad (107)$$

$$v_1 = v + ar - z_F p - h_1 p \qquad (108)$$

$$v_2 = v + ar - z_F p - h_2 p \qquad (109)$$

$$v_3 = v - br - z_R p - h_3 (p + \dot{\phi}_R) \qquad (110)^{*}$$

$$v_3 = v - br - z_R p - h_3 p \qquad (111)^{**}$$

$$v_4 = v - br - z_R p - h_4 (p + \dot{\phi}_R) \qquad (112)^{*}$$

$$v_4 = v - br - z_R p - h_4 p \qquad (113)^{**}$$

$$w_1 = w - aq + \frac{T_F}{2} p + \dot{\delta}_1 \qquad (114)$$

* Solid Rear Axle
** Split Rear Axle

$$w_2 = w - aq - \frac{T_F}{2} p + \dot{\delta}_2 \qquad (115)$$

$$w_3 = w + bq + \dot{\delta}_3 + \frac{T_R}{2} (\dot{\phi}_R + p) \qquad (116)*$$

$$w_3 = w + bq + \dot{\delta}_3 + \frac{T_R}{2} p \qquad (117)**$$

$$w_4 = w + bq + \dot{\delta}_3 - \frac{T_R}{2} (\dot{\phi}_R + p) \qquad (118)*$$

$$w_4 = w + bq + \dot{\delta}_4 - \frac{T_R}{2} p \qquad (119)**$$

The wheel velocities in the ground plane are obtained by:

$$u_{Gi} = u_i + \vartheta w_i \qquad (120)$$

$$v_{Gi} = v_i - \phi w_i \qquad (121)$$

2.15 Tire Moments

The tire-road reaction moments acting about the kingpins are computed by the following equations:

$$\begin{aligned} M_{Ti} = {} & F_{xui} \left[\overline{PT}_i \text{ SIN } \Psi_i - y_{SAi} \text{ COS } \Psi_i + h_i \left(\phi_i \text{ COS } \Psi_i \right.\right. \\ & \left.\left. - \phi_{Si} \right) \right] + F_{yui} \left[-\overline{PT}_i \text{ KK}_i \text{ COS } \Psi_i - y_{SAi} \text{ SIN } \Psi_i + h_i \right. \\ & \left. \left(\phi_i \text{ SIN } \Psi_i - \theta_{Si} \right) \right] + F_{zui} \left[-\overline{PT}_i \left(\phi_{Si} \text{ COS } \Psi_i + \right.\right. \\ & \left. \theta_{Si} \text{ SIN } \Psi_i \right) + y_{SAi} \left(\theta_{Si} \text{ COS } \Psi_i - \phi_{Si} \text{ SIN } \Psi_i \right) \\ & \left. + h_i \left(\phi_{Si} \phi_i \text{ SIN } \Psi_i - \theta_{Si} \phi_i \text{ COS } \Psi_i \right) \right] \\ & - F_{XLi} \left(Y_{SAi} + \frac{R_{WR}}{2} \right) ; \qquad i = 1,2 \end{aligned} \qquad (122)$$

where

$$\theta_{S1} = f(\delta_1, \theta_{FNT}) + \vartheta_1 \qquad (123)$$

$$\theta_{S2} = f(\delta_2, \theta_{FNT}) + \vartheta_2 \qquad (124)$$

*Solid Rear Axle
**Split Rear Axle

$$\phi_{S1} = \phi_{SA1} + \phi_1 \qquad (125)$$

$$\phi_{S2} = \phi_{SA2} + \phi_2 \qquad (126)$$

2.16 Wheel Camber with Respect to the Road

The camber angles of the wheels measured with respect to the road are given by:

$$\phi_{CGi} = \theta \text{ SIN } \Psi_i + \phi \text{ COS } \Psi_i + \phi_i \qquad i = 1,2,3,4 \qquad (127)$$

$$\phi_{CGi} = 0 \;;\; i = 3,4^*$$

2.17 Brake and Drive Torques

The drive torques generated to maintain a constant velocity are computed by:

$$\overline{TQ}_D = K_{TQ}\,(V_C - u) \quad , \quad \text{for } \overline{TQ}_D \leq TQ_{D_{MAX}} \qquad (128)$$

$$= TQ_{D_{MAX}} \quad , \quad \text{otherwise} \qquad (129)$$

where V_C is the desired velocity.

Values of 1000 in-lb/in/sec and 6000 in-lb were assigned to K_{TQ} and TQ_{DMAX} respectively. When braking is investigated, the drive torque is zero and the brake torques are step inputs with the following magnitudes:

$$\overline{TQ}_{B1} = \overline{TQ}_{B2} = 100 \text{ (PER) in-lbs} \qquad (130)$$

$$\overline{TQ}_{B3} = \overline{TQ}_{B4} = 100 \text{ (PER) in-lbs} \qquad (131)$$

where (PER) is the percent of the total brake level.

---

*Solid Rear Axle

3. NOTATION AND LIST OF SYMBOLS

3.1 Notation

The time derivative of a variable is indicated by a dot over the symbol for the variable $\left(\text{i.e., } \dot{\delta} = \frac{d\delta}{dt}, \ \ddot{\delta} = \frac{d^2\delta}{dt^2}\right)$.

Also: 1 degree = 0.0174532925 radians

1 g = 386.4 in/sec^2

The following subscript notation is employed:

F = front

R = rear, or rear axle

s = sprung mass

u = unsprung mass

i = wheel number, 1 - right front, 2 - left front, 3 - right rear, 4 - left rear

3.2 List of Symbols

a, b = distances along the vehicle fixed X axis from the sprung mass center of gravity to the spin axes of the front and rear wheels, respectively, inches.

a_i = length of steering linkage arm, inches.

a_p = length of steering arm, inches.

A_0, A_1, A_2, A_3, A_4 = constant coefficients in parabolas to represent tire cornering and camber stiffness in front tires.

$\overline{AR}$ = drive axle ratio.

Ay = $\dot{v} + ru - wp$, lateral acceleration.

B_1, B_2, B_3 = constant coefficients for curves fitted to the lateral friction coefficient properties.

C_{CR} = viscous damping coefficient of steering system connecting rod, lb-sec/in.

C_{FCR} = coulomb damping coefficient of steering system connecting rod, pounds.

C_F, C_R = viscous damping coefficient for a single wheel, effective at the wheel for the front and at the spring for the rear suspension, at the front and rear, respectively, lb-sec/in.

C_F', C_R' = coulomb damping for a single wheel, effective at the wheel for the front and at the spring for the rear suspension, at the front and rear, respectively, pounds.

F_{Ci} = longitudinal force in tire coordinate system at wheel i, pounds.

F_{Ri} = radial tire force at wheel i, pounds.

F_{SAi} = shock absorber damping force, pounds.

F_{Si} = lateral force in tire coordinate system at wheel i, pounds.

F_{XLi} = unbalanced force in front wheel due to wheel unbalance, pounds.

F_{xui}, F_{yui}, F_{zui} = components of the tire force on wheel i in the body coordinate system, pounds.

ΣF_{xu}, ΣF_{yu} = summation of the forces acting on the sprung mass along the vehicle X and Y axes, pounds.

F_{1Fi}, F_{1Ri} = coulomb damping forces in front and rear suspensions, at an individual wheel, effective at wheels in front and at spring locations in rear, pounds.

F_{2Fi}, F_{2Ri} = suspension forces produced by deflection of springs and elastic travel limits, pounds.

$f(\delta_i, \Psi_{FNT})$ = change in front wheel toe-in angles relative to the vehicle fixed coordinate axes, radians.

g = acceleration of gravity = 386.4 in/sec^2

$g(\overline{\beta}_i)$ = non-dimensional side force at wheel i.

h_i = rolling radius of wheel i, inches. (See Figure A-2)

H_i = front wheel viscous damping coefficient, in-lb/rad/sec.

i = wheel identification - 1,2,3,4 = RF, LF, RR, LR, respectively.

I_D = drive line moment of inertia for rear wheel drive, lb-sec^2-in.

I_{FW} = moment of inertia of one front wheel about the kingpin axis, lb-sec^2-in.

I_R = rear unsprung mass moment of inertia about a line through its center of gravity and parallel to the X axis, lb-sec^2-in.

I_{Wj} = rotational inertia of individual wheel at front and rear (j = F, R), respectively, lb-sec^2-in.

I_x = roll moment of inertia of sprung mass, lb-sec^2-in.

I_y = pitch moment of inertia of sprung mass, lb-sec^2-in.

I_z = yaw moment of inertia of sprung mass, lb-sec^2-in.

I_{xz} = product of inertia, lb-sec^2-in.

K_F, K_R = suspension load-deflection rate for a single wheel in the quasi-linear range about the design position, effective at the wheel for the front and at the spring for the rear suspension, at the front and rear, respectively, lb/in.

K_{RS} = roll steer gain of rear wheels relative to the vehicle coordinate system, rad/rad.

K_{SC} = steering column-gear box flexibility, in-lbs/rad.

K_T = radial tire rate in quasi-linear range for a single tire, lb/in.

K_{TQ} = gain in drive torque equation for controlling vehicle velocity, in-lbs/in/sec.

ΣM = total vehicle mass, lb-sec^2/ft

ΣM = total vehicle mass, lb-sec^2/ft.

M_{CR} = mass of steering system connecting rod, lb-sec^2/in.

M_s = sprung mass, lb-sec^2/in. (See Figure A-3)

M_{SSi} = steering torque applied by steering system connecting rod, in-lbs.

M_{Ti} = moment about the kingpin axis due to the tire-ground contact forces, in-lbs.

M_{uF} = front unsprung mass (both sides), lb-sec^2/in.

$M_1 = M_2 = \frac{M_{uF}}{2}$ = front unsprung mass at a single wheel, lb-sec^2/in. (See Figure A-3)

$M_3 = M_{uR}$ = rear unsprung mass, lb-sec^2/in. (See Figure A-3) (solid axle)

$M_3 = M_4 = \frac{M_{uR}}{2}$ = rear unsprung mass at a single wheel, lb-sec^2/in. (split axle)

M1C1, M1C2, M1C3 = constant coefficients for curves fitted to the tire-road circumferential friction coefficient properties.

N_G = gear ratio of steering gear box.

N_L = gear ratio of steering linkage.

$\left.\begin{array}{l}\Sigma N_{\varphi u}\\ \Sigma N_{\theta u}\\ \Sigma N_{\psi u}\end{array}\right\}$ = summation of moments produced by forces acting on the unsprung masses, lb-in.

$\Sigma N_{\varphi R}$ = summation of rolling moments acting on the rear axle, lb-in.

p, q, r = scalar components of sprung mass angular velocity, taken along X, Y, Z axes, respectively, rad/sec. (See Figure A-3)

$\overline{PT}$ = caster offset of front tires, inches.

$RA_0, RA_1, RA_2, RA_3, RA_4$ = constant coefficients in parabolas to represent tire cornering and camber stiffness in rear tires.

R_F, R_R = auxiliary roll stiffness (suspension stiffness in roll), at the front and rear suspensions, respectively, in-lb/rad.

$(RPS)_i$ = rotational velocity of wheel i, rad/sec.

R_w = undeflected radius of wheels, inches.

R_{WR} = width of wheel rim, inches.

S_i = total suspension force produced by the combination of springs, travel stops, viscous damping, friction, and auxiliary roll stiffness, effective at the wheel for the front suspension and at the spring location for the rear suspension, at wheel i, pounds.

$(SLIP)_i$ = longitudinal braking slip at wheel i.

T_F, T_R = track width at front and rear, respectively, inches. (See Figure A-2)

T_p = Pitman torque applied to connecting rod, in-lb.

T_S = distance between spring connections for solid rear axle, inches. (See Figure A-2)

$\overline{TQ}_{Bi}$ = brake torque at wheel i, in-lb.

$\overline{TQ}_{Di}$ = drive torque at wheel i, in-lb.

$(\overline{TQ}_D)_{max}$ = maximum drive torque, in-lb.

u, v, w = scalar components of linear velocity of the sprung mass, taken along the X, Y, Z axes, respectively, in/sec. (See Figure A-3)

$\dot{u}$, $\dot{v}$, $\dot{w}$ = scalar components of linear acceleration of the sprung mass, taken along the X, Y, Z axes, respectively, in/sec^2.

u_{Gi} = longitudinal velocity of the tire-road contact point of wheel i, in/sec.

v_{Gi} = lateral velocity of the tire-road contact point of wheel i, in/sec.

V = total vehicle velocity, in/sec.

V_c = desired vehicle velocity, in/sec.

V_i = velocity of wheel i, in/sec.

x, y, z = coordinates of vehicle center of gravity in the inertial coordinate axes system, inches.

X, Y, Z = vehicle fixed coordinate system defined as shown in Figure A-3.

Y_{CR} = connecting rod displacement, inches.

y_{SAi} = distance along the wheel spin axis from the kingpin axis to the wheel center line, inches.

z_i = distance between center of wheel i and road, inches.

z_F = static distance along the Z axis between the center of gravity of the sprung mass and the spin axis of the front wheels, inches.

z_R = static distance along the Z axis between the center of gravity of the sprung mass and the spin axis of the rear wheels, inches. (See Figure A-2)

β = vehicle body angle of sideslip, radians.

β_i = slip angle at wheel i, radians.

$\bar{\beta}_i$ = non-dimensional slip angle variable for wheel i.

β_i' = "equivalent slip angle" produced by camber at wheel i.

δ_{FWi} = front wheel steer angle produced by steering system, radians.

δ_p = Pitman shaft angle, radians.

δ_{SW} = steering wheel angle, radians.

$\delta_1, \delta_2, \delta_3$ = suspension deflection relative to the vehicle from the positions of static equilibrium, at the right front wheel center, left front wheel center, and rear axle roll center, respectively, inches. (See Figure A-3) (solid axle)

δ_3, δ_4 = suspension deflection relative to the vehicle from positions of static equilibrium, at the right rear wheel center and left rear wheel center, respectively, inches (split axle).

ϵ_{Ki} = static front wheel toe-in offset angles, radians.

ϵ_{Pi} = dead space play in connecting rod excursion, radians.

ϵ_{SP} = dead space play in steering system, radians.

ζ_3, ζ_4 = suspension deflections relative to the vehicle, from the positions of static equilibrium, measured at the right, and left rear spring positions, respectively, inches.

θ_{Si} = caster angle as a function of relative vertical deflection between wheel and body, radians.

λ_F, λ_R = term by which K_F and K_R are multiplied to represent the suspension spring rate when the suspension deflection stops are encountered.

μ_{yi} = friction coefficient for side forces.

μ_i' = circumferential coefficient of friction.

φ, θ, Ψ = Euler angles - roll, pitch, and yaw angles - defining the attitude of the sprung mass relative to the space-fixed axis system, radians.

φ_{CGi} = camber angle of wheel i relative to its tire-terrain contact plane, radians.

φ_R = angular displacement of the rear axle relative to the vehicle about a line parallel to the X-axis through the rear axle roll center (positive when clockwise as viewed from the rear), radians. (See Figure A-2)

φ_{Si} = kingpin inclination angle, radians.

φ_1, φ_2 = right front and left front wheel camber angles, respectively, relative to the vehicle-fixed coordinate axes, positive when clockwise as viewed from the rear, as a function of relative vertical deflection between wheel and body, radians.

Ψ_1, Ψ_2 = heading angle of front wheels relative to vehicle coordinate axes system, positive for clockwise steer as viewed from above vehicle, radians.

Ψ_3, Ψ_4 = heading angle of rear wheels relative to vehicle coordinate axes system, positive for clockwise steer as viewed from above vehicle, radians.

Ω_{FC}, Ω_{RC} = maximum suspension deflections in compression from the positions of static equilibrium relative to the vehicle for quasi-linear load-deflection characteristics of the springs, inches.

Ω_{FT}, Ω_{RT} = maximum suspension deflections in tension from the positions of static equilibrium relative to the vehicle for quasi-linear load-deflection characteristics of the springs, inches.

Ω_T = multiple of A_2 at which the assumed parabolic variations of small-angle cornering and camber stiffness with tire loading are constant.

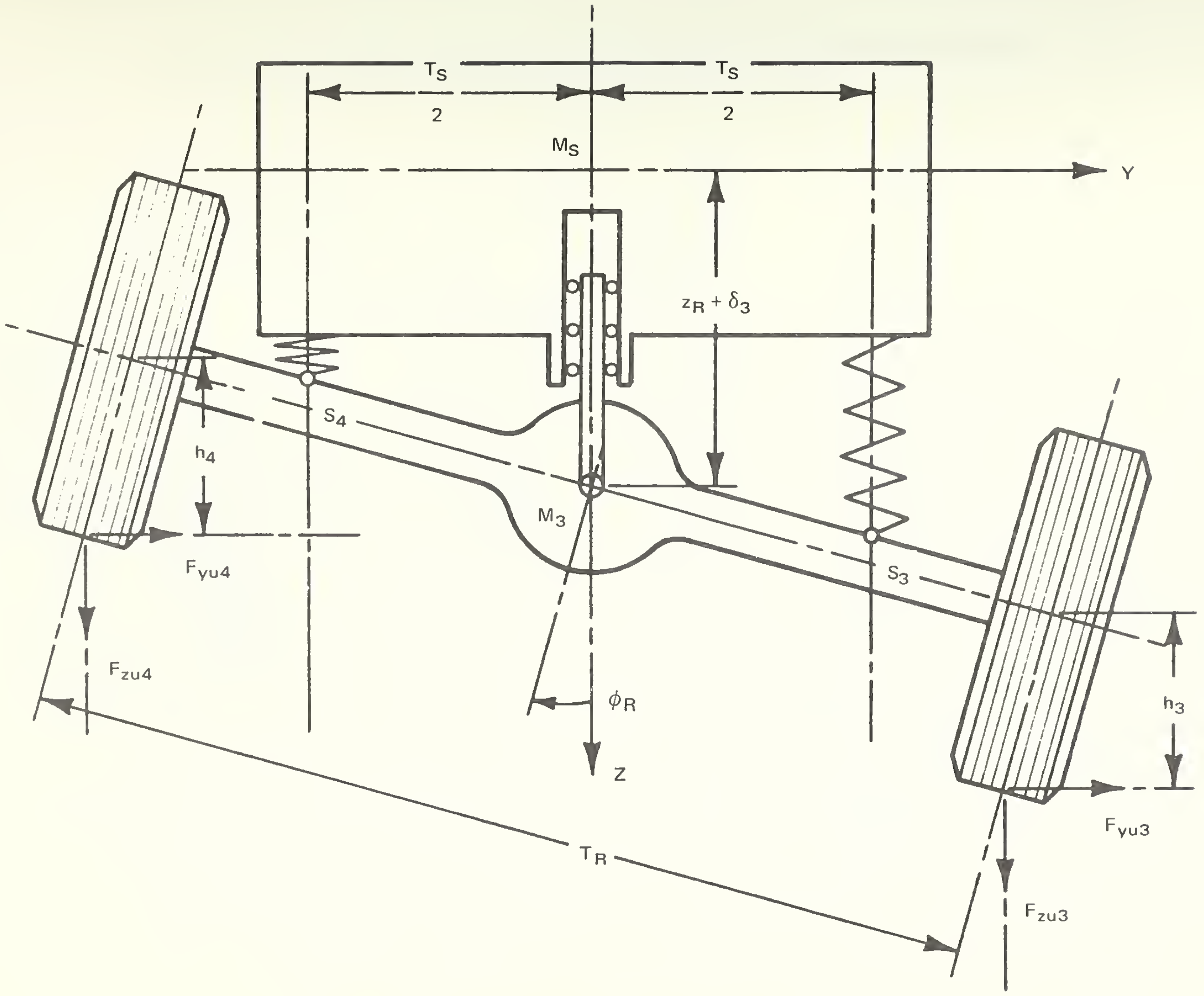


Fig. A-2 REAR AXLE REPRESENTATION

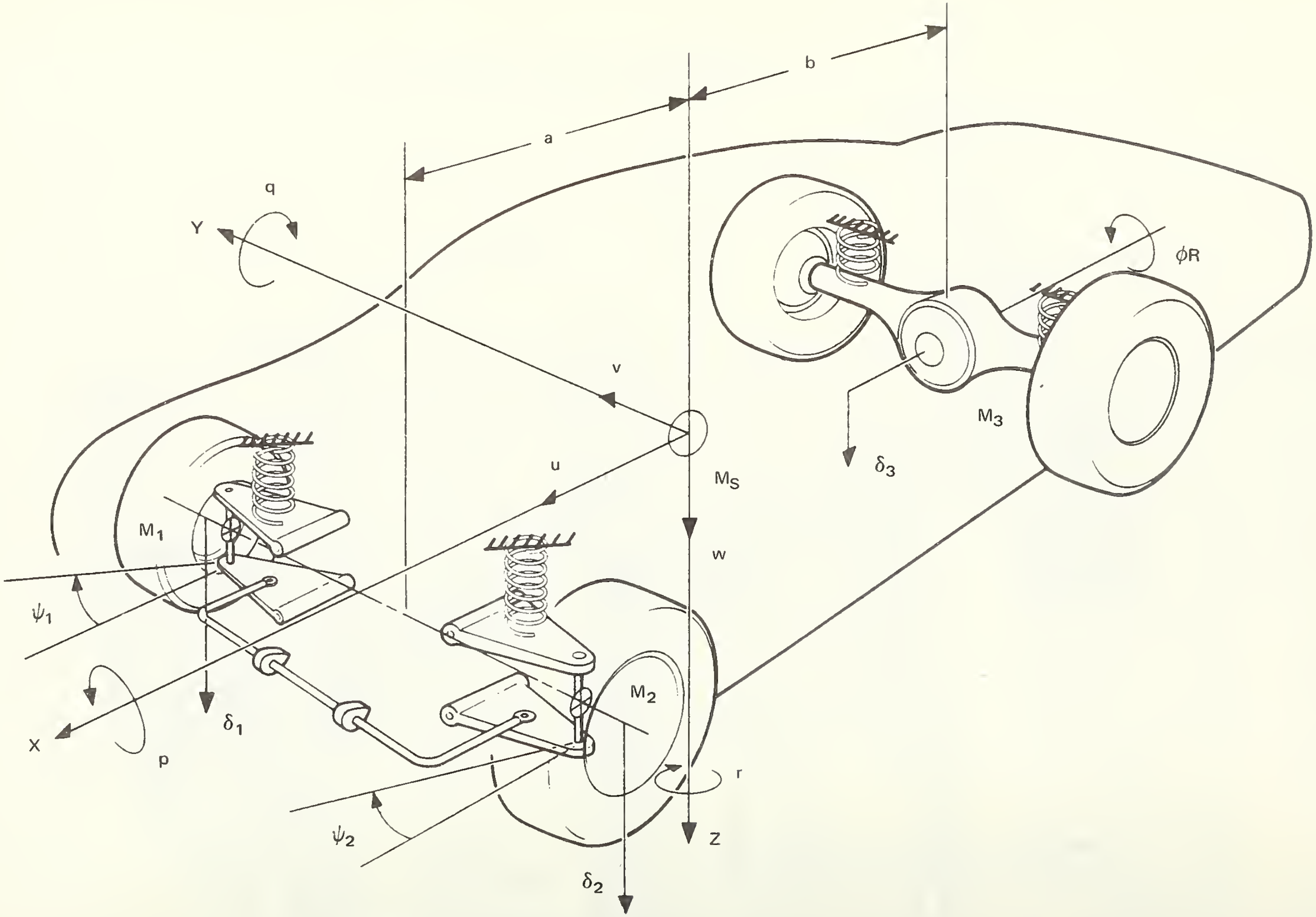


Fig. A-3 ANALYTICAL REPRESENTATION OF THE FOUR-WHEELED VEHICLE

APPENDIX B

FOUR-WHEELED VEHICLE HYBRID SIMULATION IMPLEMENTATION DOCUMENTATION

1. PRESENTED HERE IS THE COMPUTER LISTING OF THE DSL/91 DIGITAL STATIC CHECK PROGRAM

```
C          *********** LISTING OF DSL/91 DIGITAL PROGRAM **********
TITLE ### PROB:52 VEHICLE SIMULATION ###
INCON  SHAPE1=-.0065,SHAPE2=-.0037,SHAPE3=.0050 ,SHAPE4=.0029
INCON  DEL1DT=10.                  ,DEL2DT=15.   ,DEL2  = 1.866
INCON  DEL3DT=20.    ,DEL3  = 0.8  ,DEL4DT=70.   ,DEL4  =7.
INCON  THETA1=60.    ,THETA2=37.5  ,ERR1  =1.    ,ERR2  =-0.6
INCON  PHIRD =.55                  ,DFW1DT=30.   ,DFW1  =0.35
INCON  DFW2DT=37.    ,DFW2  =0.15  ,YCRDT =80.   ,YCR   =2.55
INCON  RPS1   =43.06 ,RPS2   =43.06 ,RPS3   =43.057,XPS3DT=10000.
INCON  RPS4   =43.057,XPS4DT=10000.,SEL3DT=69.   ,SEL3  =3.75
PARAM  AMS    =5.162 ,AMUF   =0.359 ,AMUR   =0.574 ,TS     = 35.86
PARAM  AIR    =800.  ,RF     =81E03 ,TF     =54.3  ,RR     =50E03
PARAM  AKT1   =812.  ,AKT2   =812.  ,AKT3   =1192. ,AKT4   =1192.
PARAM  RW     =12.85 ,AIFW   =5.815 ,AH1    =200.  ,AH2    =200.
PARAM  AMCR   =0.08  ,CFCR   =200.  ,CCR    =11.   ,AP     =6.06
PARAM  AA1    =5.53  ,AA2    =5.53  ,ANG    =17.5  ,AKSC   =610.
PARAM  ESP    =-6.   ,AKSL1 =1.17E5,AKSL2 =1.17E5,EP1    =-.2000
PARAM  EP2    =-.1   ,AIWF   =7.3777,AID    =0.3   ,ARBR   =4.125
PARAM  AIWR   =7.3777,AKF    =133.  ,AKR    =185.  ,ALAMF =5.1
PARAM  ALAMR =4.0    ,OFC    =-2.4  ,OFT    =2.3   ,ORC    =-3.1
PARAM  ORT    =2.2   ,CFP    =25.   ,CRP    =45.   ,TR     =53.3
PARAM  AKF1   =52.2  ,AKF2   =52.2  ,AKF3   =100.  ,AKF4   =100.
PARAM  AKR3   =100. ,AKR4   =100.
PARAM   DA01  =.0475, DA02 =-.7654, DA03 =.0290, DA04 =.3333
PARAM   DA05 = .2656, DA06 = .3750, DA07 =.2330 , DA08 =-.2000
PARAM   DA09 =-.1775, DA10 =.1425 , DA11 =.2666 , DA12 = .1733
PARAM   DA13 =.2166 , DA14 =.7942 , DA15 =-.2000,  DA16 =.4384
PARAM   DA17 =.0340 , DA18 =.4000                , DA21 =.0505
PARAM   DA22 =-.2625 , DA23 =.3520 , DA24 =.4630 , DA25 =.3760
PARAM   DA26 =.3860 , DA27 =.0450 , DA28 =-.37
PARAM   DA30 =-.3750, DA31 =-.4150 , DA32 =-.1180, DA36 =.2440
PARAM   DA33 =-.1181, DA34 =-.1277, DA35 =-.0950
PARAM   DA19 =+.040 , DA00=-.5678
PARAM  PHIR   =.01895,DEL1   =.3830, DA20   =.006   , DA29 = .1
*
*UNSCALED DAC VALUES FOR SYSTEM EQUATIONS
*
      DAC00  = DA00*10000.
      DAC01  = DA01*10.
      DAC02  = DA02*10000.
      DAC03  = DA03*10.
      DAC04  = DA04*20.
      DAC05  = DA05*10000.
      DAC06  = DA06*20.
      DAC07  = DA07*10000.
      DAC08  = DA08*40000.
      DAC09  = DA09*40000.
      DAC10  = DA10*10.
      DAC11  = DA11*10000.
      DAC12  = DA12*100.
      DAC13  = DA13*10000.
      DAC14  = DA14*100.
      DAC15  = DA15*100.
      DAC16  = DA16*20.
      DAC17  = DA17*10.
      DAC18  = DA18*20.
      DAC19  = DA19*10.
      DAC20  = DA20*20.
      DAC21  = DA21*20.
      DAC22   = DA22*20.
      DAC23   = DA23*1500.
      DAC24   = DA24*1500.
      DAC25   = DA25*1500.
      DAC26   = DA26*1500.
      DAC27   = DA27*20.
      DAC28   = DA28*20.
      DAC29   = DA29*20.
      DAC30   = DA30*20.
      DAC31   = DA31*20.
      DAC36   = DA36*10000.
CONTRL TSTART=0.0,FINTIM=.001,DELT=.001
DERIVATIVE REGION
*
*SYSTEM EQUATIONS
*
*FRONT WHEEL EQUATIONS OF MOTION
*
      FR1     = RHO1*AKT1
      FR2     = RHO2*AKT2
      DEL1DD = SMP/AMS - 2.*FR1/AMUF + 2.*S1P/AMUF + DAC05
      DEL2DD = SMP/AMS - 2.*FR2/AMUF + 2.*S2P/AMUF + DAC07
*
*REAR WHEELS AND AXLE COMBINATION
*
*SOLID
*
      FR3     = RHO3*AKT3
      FR4     = RHO4*AKT4
      DEL3DD = SMP/AMS + ( S3P + S4P - FR3 - FR4 ) / AMUR + DAC13
      PHIRDD =(S3P-S4P)*(TS/(2.*AIR))+(FR3/(2.*AKT3))*DAC12+...
              (FR4/(2.*AKT4))*DAC14+DAC15
*
*SPLIT
*
      SEL3DD = SMP/AMS +(2.*S3P - 2.*FR3)/AMUR + DAC11
      SEL4DD = SMP/AMS + (2.*S4P - 2.*FR4)/AMUR + DAC36
*
*SUSPENSION FORCE EQUATIONS
*
      FSA1    =33.1
      FSA2    =48.0
      FSA3    =95.3
      FSA4    =49.7
      F2F1    =20.0
      F2F2    =531.9
      F1F1    = -CFP
      F1F2    = -CFP
      F2R3    =113.9
      F2R4    =46.5
      F1R3    = -CRP
      F1R4    = -CRP
*
*ERR1,ERR2 DENOTE LIMITER SETTINGS
*
      AUXRL1 =((DEL2-DEL1)-ERR1)*(RF/(TF*TF))
      AUXRL2 =(-(DEL2-DEL1)-ERR2)*(RF/(TF*TF))
      S1P     = AUXRL1 - FSA1 - F2F1 + F1F1
      S2P     = AUXRL2    - FSA2 - F2F2 + F1F2
      S3P     = -(RR/TS)*PHIR - FSA3 - F2R3 + F1R3
      S4P    = (RR/TS)*PHIR - FSA4 - F2R4 + F1R4
      SMP    = S1P + S2P + S3P + S4P
*
*SOLID
*
      ZET3   = (TS/2.)*PHIR + DEL3
      ZET3DT = (TS/2.)*PHIRD + DEL3DT
      ZET4   = -(TS/2.)*PHIR + DEL3
      ZET4DT = -(TS/2.)*PHIRD + DEL3DT
*
*RADIAL TIRE FORCE AND ROLLING RADIUS EQUATIONS
*
      RHO1   = DEL1 + DAC01
      H1     = RW - RHO1
      RHO2   = DEL2 + DAC03
      H2     = RW - RHO2
*
*SOLID
*
      RHO3   = DEL3 + (TR/2.)*PHIR + DAC17
      RHO4   = DEL3 - (TR/2.)*PHIR + DAC19
      H3     = RW - RHO3
      H4     = RW - RHO4
*
*STEERING SYSTEM EQUATIONS
*
      DFW1DD = (1./AIFW)*(-AH1*DFW1DT - AMSS1)- DAC00
      DFW2DD = (1./AIFW)*(-AH2*DFW2DT - AMSS2)- DAC02
      YCRDD  = (1./AMCR)*(-CFCR - CCR*YCRDT + TP/AP + AMSS1/AA1 +  ...
                AMSS2/AA2)
*
*ESP,EP1,EP2 DENOTE LIMITER SETTINGS
*
      TP     = ANG*(AKSC*(DAC10      -ANG*(YCR/AP)-ESP/2.))
      AMSS1  = AKSL1*((DFW1-YCR/AA1)-EP1/2.)
      AMSS2  = AKSL2*((DFW2-YCR/AA2)-EP2/2.)
*
*WHEEL ROTATIONAL EQUATIONS
*
      RPS1DT = (-FC1*H1 - DAC08     )/AIWF
      RPS2DT = (-FC2*H2 - DAC08     )/AIWF
      DENOM = AIWR + AID*ARBR*ARBR/4.
      TERM   = AID*ARBR*ARBR/4.
      RPS3DT =(-FC3*H3 - DAC09 - TERM*XPS4DT )/DENOM
      RPS4DT =(-FC4*H4 - DAC09 - TERM*XPS3DT )/DENOM
*
*WHEEL SLIP

      SLIP1  = XI1
      SLIP2  = XI2
      SLIP3  = XI3
      SLIP4  = XI4
      XI1    = 1. - RPS1*H1/DAC24
      XI2    = 1. - RPS2*H2/DAC25
      XI3    = 1. - RPS3*H3/DAC26
      XI4    = 1. - RPS4*H4/DAC23
* ### TIRE CIRCUMFERENTIAL FORCE ###
      FC1    = -U1P*FR1
      FC2    = -U2P*FR2
      FC3    = -U3P*FR3
      FC4     = -U4P*FR4
*
* CIRCUMFERENTIAL FRICTION COEFFICIENT
*
      U1P     = SLIP1*( DAC22 + DAC04 ) + DAC20
      U2P     = SLIP2*( DAC28 + DAC06 ) + DAC21
      U3P     = SLIP3*( DAC30 + DAC16 ) + DAC27
      U4P     = SLIP4*( DAC31 + DAC18 ) + DAC29
*
*SPARE TIRES
*
      TIN     =1E-04
      THER1   =THETA1/57.3
      THER2   =THETA2/57.3
      THE1DT  =57.3*RPS1
      THE2DT  =57.3*RPS2
TERMINAL
*
*680 POTS
*REMAINING 680 POTS OBTAINED FROM IPOT SUBROUTINE
*
      P00      =DEL1DT/100.
      P18      =DEL3/10.
      P20      =4.*PHIR
      P23      =PHIRD
      P30      =DEL1/10.
      P37      =DEL2DT/100.
      P38      =DEL2/10.
      P43      =DEL3DT/100.
      P48      =SEL3DT/100.
      P57      =SEL3/10.
      P77      =DEL4DT/100.
      P78      =DEL4/10.
      P90      =THETA1/200.
      P95      =THETA2/200.
*
*
      A000    =-DEL1DT/100.
      A002    =DEL1/10.
      A005    =-DEL2DT/100.
      A007    =DEL2/10.
      A010    =-DEL3DT/100.
      A012    =DEL3/10.
      A015    =-PHIRD
      A017    =4.*PHIR
      A040    =-SEL3DT/100.
      A050    =SEL3/10.
      A080    =-DEL4DT/100.
      A085    =DEL4/10.
      A090    =THETA1/200.
      A095    =THETA2/200.
      A110    =-RPS3/100.
      A115    =-RPS4/100.
      A003    =-DEL1/10.
      A004    =SIN(THER1)
      A006    =ERR2/10.
      A008    =-H1*RPS1/DAC24
      A009    =COS(THER2)
      A011    =AUXRL2/1000.
      A014    =-COS(THER1)
```

```
A016   =-S1P/1000.
A018   =H1*RPS1/1500.
A019   =S1P/1000.
A020   =SLIP4
A021   =-S2P/1000.
A022   =RPS4DT/10000.
A023   =-FC1/(4.*AKT1)
A024   =S2P/1000.
A026   =-S3P/1000.
A028   =-(H1*FR1*U1P)/(60.*AKT1)
A029   =S3P/1000.
A030   =U3P/2.
A032   =SHAPE1
A033   =SHAPE2
A034   =SIN(THER1)
A035   =ZET4DT/100.
A036   =-ZET3/10.
A037   =SIN(THER1)
A038   =SHAPE3
A039   =SIN(THER2)
A041   =U4P/2.
A042   =SIN(THER2)
A044   =SHAPE4
A045   =-ZET4/10.
A046   =ZET3DT/100.
A048   =RHO3/2.
A049   =-H3/15.
A051   =-S4P/1000.
A053   =H2*RPS2/1500.
A054   =S4P/1000.
A055   =(S3P-S4P)/2000.
A056   =-RPS1/100.
A057   =-A039
A058   =RHO4/2.
A059   =-H4/15.
A060   =AUXRL1/1000.
A061   =-RPS2/100.
A062  =-A014
A063   =-(H2*FR2*U2P)/(60.*AKT2)
A064   =-SLIP3
A065   =-(DEL2-DEL1)/10.
A066   =SMP/1000.
A068   =-FC2/(4.*AKT2)
A069   =-A034
A070   =FRR1/10.
A072   =-SIN(THER1)
A073   =-H3*RPS3/DAC26
A075   =-((SLIP4*DAC18+DAC29)/20.)
A076   =RPS3DT/10000.
A077   =COS(THER1)
A078   =H3*RPS3/1500.
A081   =SLIP2
A082   =-SIN(THER2)
A083   =-(H3*FR3*U3P)/(60.*AKT3)
A084   =-H2/15.
A086   =-RHO1/2.
A087   =COS(THER2)
A088   =-FC3/(4.*AKT3)
A089   =-H1/15.
A091   =U1P/2.
```

```
      A092   =SLIP3
      A093   =-H2*RPS2/DAC25
      A094   =-((SLIP3*DAC16+DAC27)/20.)
      A096   =-RH02/2.
      A097   =-SLIP4
      A099   =SIN(THER2)
      A100   =RPS1/100.
      A101   =SLIP1
      A102   =-SLIP1
      A103   =-H4*RPS4/DAC23
      A104   =-((SLIP1*DAC04+DAC20)/20.)
      A106   =-RH03/2.
      A105   =RPS2/100.
      A108   =H4*RPS4/1500.
      A109   =-((SLIP2*DAC06+DAC21)/20.)
      A111   =U2P/2.
      A113   =-(H4*FR4*U4P)/(60.*AKT4)
      A114   =-SLIP2
      A116   =-RH04/2.
      A117   =-A087
      A118   =-FC4/(4.*AKT4)
*
*
      D000   =DEL1DD/1000.
      D002   =-DEL1DT/100.
      D005   =DEL2DD/1000.
      D007   =-DEL2DT/100.
      D010   =DEL3DD/1000.
      D012   =-DEL3DT/100.
      D015   =PHIRDD/10.
      D017   =-PHIRD/2.5
      D040   =SEL3DD/1000.
      D050   =-SEL3DT/100.
      D080   =SEL4DD/1000.
      D085   =-DEL4DT/100.
      D090   =THE1DT/2000.
      D095   =THE2DT/2000.
      D100   =-RPS1DT/1000.
      D105   =-RPS2DT/1000.
      D110   =RPS3DT/1000.
      D115   =RPS4DT/1000.
*
* 231-R POTS
*
      Q200    =AKF1*10.0*TIN
      Q201    =.5186
      Q202   = 1.0/(20.0*AMCR)
      Q203    =.6188
      Q204   = 2.0/3.0
      Q205    =YCRDT/200.
      Q206    =YCR/3.
      Q207 = .2000
      Q208    =DFW2DT/100.
      Q209    =2.*DFW2
      Q210     =AKR3*10.0*TIN
      Q211     =.7692
      Q212   = AH2/(100.0*AIFW)
      Q213     =.7803
      Q214   = AKSL2/(40000.0*AIFW)
      Q215     =AKR4*10.0*TIN
```

```
     Q216      =.7692
     Q217    = 3.0*ANG/(10.0*AP)                                                 00008700
     Q218      =.7803
     Q219    = AKSC*ANG/(AP*2000.0)                                              00008680
     Q220      =AKF2*10.0*TIN
     Q221      =.5186
     Q222    = AH1/(100.0*AIFW)                                                  00008790
     Q223     =.6188
     Q224    = AKSL1/(40000.0*AIFW)                                              00008770
     Q235     =DFW1DT/100.
     Q236     =2.*DFW1
     Q255    =.8056
     Q256    =.8066
     Q257    = CFCR/20000.0                                                      00008690
     Q258    = CCR/100.0                                                         00008670
     Q265    =.7652
     Q266    =.8186
     Q267    = AKSL2/(AA2*40000.0)                                               00008660
     Q268    =3.0/AA2                                                            00008710
     Q275    =.8056
     Q276    =.8066
     Q285    =.7652
     Q286    =.8186
     Q287    =3.0/AA1                                                            00008830
     Q288    = AKSL1/(AA1*40000.0)                                               00008650
*
*231-R AMP'S
*
     A250    =YCRDT/200.
     A251    =-YCR/3.
     A260    =-DFW2DT/100.
     A261    =2.*DFW2
     A280    =-DFW1DT/100.
     A281    =2.*DFW1
     A200    =DFL1/10.
     A201    =-F2F1/1000.
     A210    =ZFT3/10.
     A211    =-F2R3/1000.
     A215    =ZFT4/10.
     A216    =-F2R4/1000.
     A220    =DFL2/10.
     A221    =-F2F2/1000.
     A227    =-.2*DFW2
     A230    =-FSA1/1000.
     A231    =-FSA2/1000.
     A232    =-FSA3/1000.
     A233    =-FSA4/1000.
     A237    =-.2*DFW1
     A240    =DFL1DT/100.
     A241    =DFL2DT/100.
     A242    =-ZFT3DT/100.
     A243    =ZFT3DT/100.
     A244    =-ZFT4DT/100.
     A245    =ZFT4DT/100.
     A252    =-(AMCR*YCRDD/20000.)
     A253    =-CFCR/20000.
     A254    =CFCR/20000.
     A262    =-2.*(DFW2-(YCR/AA2))
     A263    =2.*AMSS2/AKSL2
     A264    =-A263
```

```
          A270    =-((DAC10-(ANG*YCR)/AP)/10.)
          A271    =(FSP/2.)/10.
          A272    =-(A270+A271)
          A273    =-A250
          A282    =-2.*(DFW1-(YCR/AA1))
          A283    =2.*AMSS1/AKSL1
          A284    =-A283
          A292    =2.*(EP1/2.)
          A293    =2.*(EP2/2.)
 *
 *231 DERIVATIVES
 *
          D250    =YCRDD/20000.
          D251    =-YCRDT/30.
          D260    =-DFW2DD/1000.
          D261    =DFW2DT/5.
          D280    =-DFW1DD/1000.
          D281    =DFW1DT/5.
          T220    =A230
          T222    =A232
          T223    =A233
          T224    =A201
          T225    =A221
          T226    =A211
          T227    =A216
          T228    =A231
          T271    =A237
          T272    =A227
          DUMMY  =  DEBUG(1.,0.)
          CALL PUNCH
END
STOP
```

```
               *** DSL/91  SIMULATION DATA ***                                                          72.285        16:03:38.23
          TITLE === PROB052 VEHICLE SIMULATION ===
          INCON   SHAPE1=-.0065,SHAPE2=-.0037,SHAPE3=.0050 ,SHAPE4=.0029
          INCON   DEL1DT=10.                   ,DEL2DT=15.    ,DEL2  = 1.866
          INCON   DEL3DT=20.    ,DEL3  = 0.8   ,DEL4DT=70.    ,DEL4  =7.
          INCON   THETA1=60.    ,THETA2=37.5   ,ERR1  =1.     ,ERR2  =-0.6
          INCON   PHIRD =.55                   ,DFW1DT=30.    ,DFW1  =0.35
          INCON   DFW2DT=37.    ,DFW2  =0.15   ,YCRDT =80.    ,YCR   =2.55
          INCON   RPSI  =43.06  ,RPS2  =43.06  ,RPS3  =43.057,XPS3DT=10000.
          INCON   RPS4  =43.057,XPS4DT=10000.,SEL3DT=69.     ,SEL3  =3.75
          PARAM   AMS   =5.162  ,AMUF  =0.359  ,AMUR  =0.574  ,TS    = 35.86
          PARAM   AIR   =800.   ,RF    =81E03  ,TF    =54.3   ,RR    =50E03
          PARAM   AKT1  =812.   ,AKT2  =812.   ,AKT3  =1192.  ,AKT4  =1192.
          PARAM   RW    =12.85  ,AIFW  =5.815  ,AH1   =200.   ,AH2   =200.
          PARAM   AMCR  =0.08   ,CFCR  =200.   ,CCR   =11.    ,AP    =6.06
          PARAM   AA1   =5.53   ,AA2   =5.53   ,ANG   =17.5   ,AKSC  =610.
          PARAM   ESP   =-6.    ,AKSL1 =1.17E5,AKSL2 =1.17E5,EP1   =-.2000
          PARAM   EP2   =-.1    ,AIWF  =7.3777,AID   =0.3    ,ARBR  =4.125
          PARAM   AIWR  =7.3777,AKF   =133.   ,AKR   =185.   ,ALAMF =5.1
          PARAM   ALAMR =4.0    ,OFC   =-2.4   ,OFT   =2.3    ,ORC   =-3.1
          PARAM   ORT   =2.2    ,CFP   =25.    ,CRP   =45.    .TR    =53.3
          PARAM   AKFI  =52.2   ,AKF2  =52.2   ,AKF3  =100.   ,AKF4  =100.
          PARAM   AKR3  =100. ,AKR4  =100.
          PARAM    DA01  =.0475, DA02 =-.7654, DA03 =.0290, DA04 =.3333
          PARAM    DA05 = .2656, DA06 = .3750, DA07 =.2330 , DA08 =-.2000
          PARAM    DA09 =-.1775, DA10 =.1425 , DAI1 =.2666 , DA12 = .1733
          PARAM    DA13 =.2166 , DA14 =.7942 , DA15 =-.2000,  DA16 =.4384
          PARAM    DA17 =.0340 , DA18 =.4000                , DA21 =.0505
          PARAM    DA22 =-.2625 , DA23 =.3520 , DA24 =.4630 , DA25 =.3760
          PARAM    DA26 =.3860 , DA27 =.0450 , DA28 =-.37
          PARAM    DA30 =-.3750, DA31 =-.4150 , DA32 =-.1180, DA36 =.2440
          PARAM    DA33 =-.1181, DA34 =-.1277, DA35 =-.0950
          PARAM    DA19 =+.040 , DA00=-.5678
          PARAM   PHIR  =.01895,DEL1  =.3830, DA20  =.006   , DA29 = .1
          CONTRL TSTART=0.0,FINTIM=.001,DELT=.001
          END

***DSL MESSAGE 20***   NO OUTPUT REQUESTED...WARNING ONLY.

DSL/91  SIMULATION TIME=   0.0  SECONDS.                                                               72.285        16:03:39.31

 DEBUG OUTPUT, BLOCK   3 AT TIME=  0.1000E-02


 TIME       1.0000E-03  DELT       1.0000E-03  DELMIN      0.0          DELMAX     7.2370E 75  TSTART      0.0          FINTIM     1.0000E-03
 CLKTIM     0.0         NALARM     0.0         DELS        0.0          DELNIX     7.2370E 75  DELADC      7.2370E 75  DELDAC     7.2370E 75
 DELSTP     7.2370E 75  DELMRK     7.2370E 75  SHAPE1     -6.5000E-03   SHAPE2    -3.7000E-03  SHAPE3      5.0000E-03  SHAPE4     2.9000E-03
 DEL1DT     1.0000E 01  DEL2DT     1.5000E 01  DEL2        1.8660E 00   DEL3DT     2.0000E 01  DEL3        8.0000E-01  DEL4DT     7.0000E 01
 DEL4       7.0000E 00  THETAI     6.0000E 01  THETA2      3.7500E 01   ERR1       1.0000E 00  ERR2       -6.0000E-01  PHIRD      5.5000E-01
 DFWIDT     3.0000E 01  DFW1       3.5000E-01  DFW2DT      3.7000E 01   DFW2       1.5000E-01  YCRDT       8.0000E 01  YCR        2.5500E 00
 RPSI       4.3060E 01  RPS2       4.3060E 01  RPS3        4.3057E 01   XPS3DT     1.0000E 04  RPS4        4.3057E 01  XPS4DT     1.0000E 04
 SEL3DT     6.9000E 01  SEL3       3.7500E 00  AMS         5.1620E 00   AMUF  -    3.5900E-01 .AMUR       5.7400E-01  TS         3.5860E 01
 AIR        8.0000E 02  RF         8.1000E 04  TF          5.4300E 01   RR         5.0000E 04  AKT1        8.1200E 02  AKT2       8.1200E 02
 AKT3       1.1920E 03  AKT4       1.1920E 03  RW          1.2850E 01   AIFW       5.8150E 00  AH1         2.0000E 02  AH2 ,      2.0000E 02
 AMCR       8.0000E-02  CFCR       2.0000E 02  CCR         1.1000E 01   AP         6.0600E 00  AA1         5.5300E 00  AA2        5.5300E 00
 ANG        1.7500E 01  AKSC       6.1000E 02  ESP        -6.0000E 00   AKSL1      1.1700E 05  AKSL2       1.1700E 05  EP1       -2.0000E-01
 EP2       -1.0000E-01  AIWF       7.3777E 00  AID         3.0000E-01   ARBR       4.1250E 00  AIWR        7.3777E 00  AKF        1.3300E 02
 AKR        1.8500E 02  ALAMF      5.1000E 00  ALAMR       4.0000E 00   DFC       -2.4000E 00  OFT         2.3000E 00  DRC       -3.1000E 00
 ORT        2.2000E 00  CFP        2.5000E 01  CRP         4.5000E 01   TR         5.3300E 01  AKF1        5.2200E 01  AKF2       5.2200E 01
 AKF3       1.0000E 02  AKF4       1.0000E 02  AKR3        1.0000E 02   AKR4       1.0000E 02  DA01        4.7500E-02  DA02      -7.6540E-01
 DA03       2.9000E-02  DA04       3.3330E-01  DA05        2.6560E-01   DA06       3.7500E-01  DA07        2.3300E-01  DA08      -2.0000E-01
```

```
DA09     -1.7750E-01  DA10      1.4250E-01  DA11      2.6660E-01  DA12      1.7330E-01  DA13      2.1660E-01  DA14      7.9420E-01
DA15     -2.0000E-01  DA16      4.3840E-01  DA17      3.4000E-02  DA18      4.0000E-01  DA21      5.0500E-02  DA22     -2.6250E-01
DA23      3.5200E-01  DA24      4.6300E-01  DA25      3.7600E-01  DA26      3.8600E-01  DA27      4.5000E-02  DA28     -3.7000E-01
DA30     -3.7500E-01  DA31     -4.1500E-01  DA32     -1.1800E-01  DA36      2.4400E-01  DA33     -1.1810E-01  DA34     -1.2770E-01
DA35     -9.5000E-02  DA19      4.0000E-02  DA00     -5.6780E-01  PHIR      1.8950E-02  DEL1      3.8300E-01  DA20      6.0000E-03
DA29      1.0000E-01  DAC00    -5.6780E 03  DAC01     4.7500E-01  DAC02    -7.6540E 03  DAC03     2.9000E-01  DAC04     6.6660E 00
DAC05     2.6560E 03  DAC06     7.5000E 00  DAC07     2.3300E 03  DAC08    -8.0000E 03  DAC09    -7.1000E 03  DAC10     1.4250E 00
DAC11     2.6660E 03  DAC12     1.7330E 01  DAC13     2.1660E 03  DAC14     7.9420E 01  DAC15    -2.0000E 01  DAC16     8.7680E 00
DAC17     3.4000E-01  DAC18     8.0000E 00  DAC19     4.0000E-01  DAC20     1.2000E-01  DAC21     1.0100E 00  DAC22    -5.2500E 00
DAC23     5.2800E 02  DAC24     6.9450E 02  DAC25     5.6400E 02  DAC26     5.7900E 02  DAC27     9.0000E-01  DAC28    -7.4000E 00
DAC29     2.0000E 00  DAC30    -7.5000E 00  DAC31    -8.3000E 00  DAC36     2.4400E 03  FR1       6.9670E 02  FR2       1.7507E 03
DEL1DD   -1.7975E 03  DEL2DD   -1.1139E 04  FR3       1.9609E 03  FR4       8.2842E 02  DEL3DD   -3.5933E 03  PHIRDD    1.8135E 01
SEL3DD   -5.3551E 03  SEL4DD   -1.0574E 03  FSA1      3.3100E 01  FSA2      4.8000E 01  FSA3      9.5300E 01  FSA4      4.9700E 01
F2F1      2.0000E 01  F2F2      5.3190E 02  F1F1     -2.5000E 01  F1F2     -2.5000E 01  F2R3      1.1390E 02  F2R4      4.6500E 01
F1R3     -4.5000E 01  F1R4     -4.5000E 01  AUXRL1    1.3269E 01  AUXRL2   -2.4257E 01  S1P      -6.4831E 01  S2P      -6.2916E 02
S3P      -2.8062E 02  S4P      -1.1478E 02  SMP      -1.0894E 03  ZET3      1.1398E 00  ZET3DT    2.9861E 01  ZET4      4.6023E-01
ZET4DT    1.0139E 01  RHO1      8.5800E-01  H1        1.1992E 01  RHO2      2.1560E 00  H2        1.0694E 01  RHO3      1.6450E 00
RHO4      6.9498E-01  H3        1.1205E 01  H4        1.2155E 01  DFW1DD    4.8699E 03  DFW2DD    1.1635E 04  YCRDD    -1.5021E 05
TP       -3.1372E 04  AMSS1    -1.3012E 03  AMSS2    -3.0551E 04  RPS1DT    1.6315E 03  RPS2DT    3.6939E 03  DENOM     8.6539E 00
TERM      1.2762E 00  RPS3DT    2.1676E 03  RPS4DT    1.6698E 03  SLIP1     2.5648E-01  SLIP2     1.8354E-01  SLIP3     1.6675E-01
SLIP4     8.7900E-03  XI1       2.5648E-01  XI2       1.8354E-01  XI3       1.6675E-01  XI4       8.7900E-03  FC1      -3.3662E 02
FC2      -1.8003E 03  FC3      -2.1794E 03  FC4      -1.6547E 03  U1P       4.8317E-01  U2P       1.0284E 00  U3P       1.1114E 00
U4P       1.9974E 00  TIN       1.0000E-04  THER1     1.0471E 00  THER2     6.5445E-01  THE1DT    2.4673E 03  THE2DT    2.4673E 03
ZZ0001    0.0         ZZ0002    0.0         P00       1.0000E-01  P18       8.0000E-02  P20       7.5800E-02  P23       5.5000E-01
P30       3.8300E-02  P37       1.5000E-01  P38       1.8660E-01  P43       2.0000E-01  P48       6.9000E-01  P57       3.7500E-01
P77       7.0000E-01  P78       7.0000E-01  P90       3.0000E-01  P95       1.8750E-01  A000     -1.0000E-01  A002      3.8300E-02
A005     -1.5000E-01  A007      1.8660E-01  A010     -2.0000E-01  A012      8.0000E-02  A015     -5.5000E-01  A017      7.5800E-02
A040     -6.9000E-01  A050      3.7500E-01  A080     -7.0000E-01  A085      7.0000E-01  A090      3.0000E-01  A095      1.8750E-01
A110     -4.3057E-01  A115     -4.3057E-01  A003     -3.8300E-02  A004      8.6599E-01  A006     -6.0000E-02  A008     -7.4352E-01
A009      7.9338E-01  A011     -2.4257E-02  A014     -5.0007E-01  A016      6.4831E-02  A018      3.4425E-01  A019     -6.4831E-02
A020      8.7900E-03  A021      6.2916E-01  A022      1.6698E-01  A023      1.0364E-01  A024     -6.2916E-01  A026      2.8062E-01
A028     -8.2857E-02  A029     -2.8062E-01  A030      5.5572E-01  A032     -6.5000E-03  A033     -3.7000E-03  A034      8.6599E-01
A035      1.0139E-01  A036     -1.1398E-01  A037      8.6599E-01  A038      5.0000E-03  A039      6.0872E-01  A041      9.9868E-01
A042      6.0872E-01  A044      2.9000E-03  A045     -4.6023E-02  A046      2.9861E-01  A048      8.2251E-01  A049     -7.4700E-01
A051      1.1478E-01  A053      3.0699E-01  A054     -1.1478E-01  A055     -8.2922E-02  A056     -4.3060E-01  A057     -6.0872E-01
A058      3.4749E-01  A059     -8.1033E-01  A060      1.3269E-02  A061     -4.3060E-01  A062      5.0007E-01  A063     -3.9517E-01
A064     -1.6675E-01  A065     -1.4830E-01  A066     -1.0894E 00  A068      5.5428E-01  A069     -8.6599E-01  A070      1.0000E-01
A072     -8.6599E-01  A073     -8.3325E-01  A075     -1.0352E-01  A076      2.1676E-01  A077      5.0007E-01  A078      3.2164E-01
A081      1.8354E-01  A082     -6.0872E-01  A083     -3.4144E-01  A084     -7.1293E-01  A086     -4.2900E-01  A087      7.9338E-01
A088      4.5708E-01  A089     -7.9947E-01  A091      2.4159E-01  A092      1.6675E-01  A093     -8.1646E-01  A094     -1.1810E-01
A096     -1.0780E 00  A097     -8.7900E-03  A099      6.0872E-01  A100      4.3060E-01  A101      2.5648E-01  A102     -2.5648E-01
A103     -9.9121E-01  A104     -9.1484E-02  A106     -8.2251E-01  A105      4.3060E-01  A108      3.4891E-01  A109     -1.1933E-01
A111      5.1418E-01  A113     -2.8121E-01  A114     -1.8354E-01  A116     -3.4749E-01  A117     -7.9338E-01  A118      3.4703E-01
D000     -1.7975E 00  D002     -1.0000E-01  D005     -1.1139E 01  D007     -1.5000E-01  D010     -3.5933E 00  D012     -2.0000E-01
D015      1.8135E 00  D017     -2.2000E-01  D040     -5.3551E 00  D050     -6.9000E-01  D080     -1.0574E 00  D085     -7.0000E-01
D090      1.2337E 00  D095      1.2337E 00  D100     -1.6315E 00  D105     -3.6939E 00  D110      2.1676E 00  D115      1.6698E 00
Q200      5.2200E-02  Q201      5.1860E-01  Q202      6.2500E-01  Q203      6.1880E-01  Q204      6.6667E-01  Q205      4.0000E-01
Q206      8.5000E-01  Q207      2.0000E-01  Q208      3.7000E-01  Q209      3.0000E-01  Q210      1.0000E-01  Q211      7.6920E-01
Q212      3.4394E-01  Q213      7.8030E-01  Q214      5.0301E-01  Q215      1.0000E-01  Q216      7.6920E-01  Q217      8.6634E-01
Q218      7.8030E-01  Q219      8.8077E-01  Q220      5.2200E-02  Q221      5.1860E-01  Q222      3.4394E-01  Q223      6.1880E-01
Q224      5.0301E-01  Q235      3.0000E-01  Q236      7.0000E-01  Q255      8.0560E-01  Q256      8.0660E-01  Q257      1.0000E-02
Q258      1.1000E-01  Q265      7.6520E-01  Q266      8.1860E-01  Q267      5.2893E-01  Q268      5.4250E-01  Q275      8.0560E-01
Q276      8.0660E-01  Q285      7.6520E-01  Q286      8.1860E-01  Q287      5.4250E-01  Q288      5.2893E-01  A250      4.0000E-01
A251     -8.5000E-01  A260     -3.7000E-01  A261      3.0000E-01  A280     -3.0000E-01  A281      7.0000E-01  A200      3.8300E-02
A201     -2.0000E-02  A210      1.1398E-01  A211     -1.1390E-01  A215      4.6023E-02  A216     -4.6500E-02  A220      1.8660E-01
A221     -5.3190E-01  A227     -3.0000E-02  A230     -3.3100E-02  A231     -4.8000E-02  A232     -9.5300E-02  A233     -4.9700E-02
A237     -7.0000E-02  A240      1.0000E-01  A241      1.5000E-01  A242     -2.9861E-01  A243      2.9861E-01  A244     -1.0139E-01
A245      1.0139E-01  A252      6.0084E-01  A253     -1.0000E-02  A254      1.0000E-02  A262      6.2224E-01  A263     -5.2224E-01
A264      5.2224E-01  A270      5.9389E-01  A271     -3.0000E-01  A272     -2.9389E-01  A273     -4.0000E-01  A282      2.2224E-01
A283     -2.2242E-02  A294      2.2242E-02  A292     -2.0000E-01  A293     -1.0000E-01  D250     -7.5105E 00  D251     -2.6667E 00
D260     -1.1635E 01  D261      7.4000E 00  D280     -4.8699E 00  D281      6.0000E 00  T220     -3.3100E-02  T222     -9.5300E-02
T223     -4.9700E-02  T224     -2.0000E-02  T225     -5.3190E-01  T226     -1.1390E-01  T227     -4.6500E-02  T228     -4.8000E-02
T271     -7.0000E-02  T272     -3.0000E-02  DUMMY     0.0         ZZ0003    0.0
```

2. PRESENTED HERE IS THE COMPUTER LISTING OF THE IBM 360/91 DIGITAL PROGRAM

```
C        ********** LISTING OF IBM 360/91 DIGITAL PROGRAM **********
C                                                                       00002500
C      VEHICLE HANDLING  MODEL C
C      VEHICLE HANDLING  MODEL C                                        00002510
C                                                                       00002520
      COMMON/APL/ OPEN  ,RTSW  ,LDTSW ,RBSW
      COMMON/DEVICE/KEYBD,ITTY,ICDRD,LPTR
      COMMON /ECBBLK/PILECB,TCNECB,TIMECB,ADAECB,TDAECB
      COMMON /ECBBLK/AD2ECB,AD1ECB,CLSECB,CLRECB,ICECB ,OPECB
      COMMON /ECBBLK/OSECB ,DONECB
      COMMON/SPLTAX/SPSR3,CPSR3,SPSR4,CPSR4,SCR3,SCR4,TBCR3,TBCR4,TBSR3,00002530
     1   TBSR4,TRSR3,TRSR4,TRCR3,TRCR4,PSR3,PSR4,IAX                    00002540
      COMMON/NEWTBS/TQBF(50),PBF(50),TQBR(50),PBR(50),AMA(50),AFRI(50), 00002550
     1   AMB(50),GAMM(50),AFA(50),GAMF(50),AFB(50),SLPP(50),NTF,NTR,    00002560
     1   NMA,NMB,NFA,NFB                                                00002570
      COMMON/DELS/DELSWC                                                00002580
      COMMON/EXTRA/            UOUT,ROUT,       PSIOUT,EXTAB(100),      00002590
     1   DLYTB(100),RTAB(100)                                           00002600
      COMMON/ALPHA/ALPH(20)                                             00002610
      COMMON/DDD/DIN,DDTO,DDT,TIME5,DDO,DD,DOUT                         00002620
      COMMON/XX/IDAC(48),IADC(32)                                       00002630
      COMMON/INOUT/INA(32),IOUTA(48),IN(32),IOUT(48),ISW1,ISW7,SFIN(32),00002640
     1     SFOUT(48),IPRT,ITMP(48)                                      00002650
      COMMON/EFFS/ANUM,ADEN,ANUMDT,ADENDT,ANUMO,ADENO,ANUMDO,ADENDO,    00002660
     1   ANOUT,ADOUT                                                    00002670
      COMMON/NUTS/TOO                                                   00002680
      COMMON/ANALOG/DEL1DT,DEL2DT,DEL3DT,DEL1,DEL2,DEL3,PHIRD,PHIR,     00002690
     1    DELFW1,DELFW2,U1P,U2P,U3P,U4P,S1P,S2P,S3P,S4P,TTA,PFSLP       00002700
      COMMON/VARS/P,Q,R,U,V,W,X,Y,Z,THE,PHI,PSI,PO,QO,RO,UO,VO,WO,XO,   00002710
     1    YO,ZO,THEO,PHIO,PSIO                                          00002720
      COMMON/NEWER/TIME25,TIME10,RMAX,ETAMAX,PSI5,PHIMAX,DSWMAX         00002730
C     COMMON/THINGS/TMAX1,TMAX2,TMAX3,TQRMAX,TQFMAX,PSIMAX              00012300
      COMMON/THINGS/TMAX1,TMAX2,TMAX3,TQRMAX,TQFMAX,PSIMAX,ONER
      COMMON/TABLES/PHIENT(7),THEENT(7),PSIENT(7),PHIRR(7),THERR(7),    00002750
     1   PSIRR(7)                                                       00002760
      COMMON/FRIDAY/BTVMAX,BTVTB(100)                                   00002770
      COMMON/COMBLK/AIXP,SM,AIYP,AIXZP,GAM1,GAM2,GAM3,AIXBR,AIYBR,      00002780
     1    AIZBR,A1,A2,AIXZBR,A12,E1,E2,E3,DELTA,GV1,GV2,GP1,GP2,GR1,    00002790
     1    GR2,CIP,CIVP,RZF,RZR,A2T,CA20,CA23,TIME,DT,     ANGNL,ANGNLO  00002800
     1    ,TRO2,TFO2,TSO2,G,THRD,TWN7                                   00002810
      COMMON/SP7BLK/N1,N2,IPOT(244),IPOTAD(244),PARAM(300)              00002820
      COMMON/MICKEY/IRUNTB(100),VTB(100),SNLTB(100),SNRTB(100),DSWTB(10000002830
     1  ),BTTB(100),DTB(100),EFFTB(100),DYTB(100),PSITB (100),NRUN,     00002840
     1   YSPEC,PSIM,XPF                                                 00002850
     1   ,TMPTB(100)                                                    00002860
      COMMON/NONAME/XEND,O,EXIT2
      DIMENSION  ADC1(24),ADC2(4)
      EQUIVALENCE (ADC1(24),IN(24))      ,(ADC2(1),IN(25))
      EQUIVALENCE                                                       00002880
     1   (PARAM( 1),AMS)     , (PARAM( 2),AMUF)  , (PARAM( 3),AMUR)   , 00002890
     1   (PARAM( 4),ZF)      , (PARAM( 5),ZR)    , (PARAM( 6),A)      , 00002900
     1   (PARAM( 7),B)       , (PARAM( 8),TF)    , (PARAM( 9),TR)     , 00002910
     1   (PARAM(10),TS)      , (PARAM(11),AIX)   , (PARAM(12),AIY)    , 00002920
     1   (PARAM(13),AIZ)     , (PARAM(14),AIXZ)  , (PARAM(15),AIR)    , 00002930
     1   (PARAM(16),CF)      , (PARAM(17),RF)    , (PARAM(18),CFP)    , 00002940
     1   (PARAM(19),AKF)     , (PARAM(20),ALAMF) , (PARAM(21),OFC)    , 00002950
     1   (PARAM(22),OFT)     , (PARAM(23),CR)    , (PARAM(24),RR)     , 00002960
     1   (PARAM(25),CRP)     , (PARAM(26),AKR)   , (PARAM(27),ALAMR)  , 00002970
     1   (PARAM(28),ORC)     , (PARAM(29),ORT)   , (PARAM(30),AKRS)   , 00002980
     1   (PARAM(31),RW)      , (PARAM(32),B4)     , (PARAM(33),OT)     ,
     1   (PARAM(34),CA0)     , (PARAM(35),CA1)    , (PARAM(36),CA2)    ,
     1   (PARAM(37),CA3)     , (PARAM(38),CA4)    , (PARAM(39),AISW)   ,
     1   (PARAM(44),AKDL)    , (PARAM(41),AKSC)   , (PARAM(42),ANG)    ,
     1   (PARAM(43),WG)      , (PARAM(40),ANL2)   , (PARAM(45),AKSL)
      EQUIVALENCE
     1   (PARAM(46),ANL1)    , (PARAM(47),AIFW)   , (PARAM(48),HDL)    ,
     1   (PARAM(49),AIWF)    , (PARAM(50),AIWR)   , (PARAM(51),AID)    ,
     1   (PARAM(52),ARBR)    , (PARAM(53),EPS1)   , (PARAM(54),EPS2)   ,
     1   (PARAM(55),PTBR)    , (PARAM(56),YSA1)   , (PARAM(57),YSA2)   ,
     1   (PARAM(58),YHS1)    , (PARAM(59),YHS2)   , (PARAM(60),AKD)    ,
     1   (PARAM(61),TQDBR)   , (PARAM(62),AK)     , (PARAM(63),PIN)    ,
     1   (PARAM(64),QIN)     , (PARAM(65),RIN)    , (PARAM(66),UIZ)    ,
     1   (PARAM(67),VIN)     , (PARAM(68),WIN)    , (PARAM(69),XIN)    ,
     1   (PARAM(70),YIN)     , (PARAM(71),ZIN)    , (PARAM(72),THEIN)  ,
     1   (PARAM(73),PHIIN)   , (PARAM(74),PSIIN)  , (PARAM(75),DTIN)   ,
     1   (PARAM(76),TEND)    , (PARAM(77),AKT1)   , (PARAM(78),AKT2)  ,
     1   (PARAM(79),AKT3)    , (PARAM(80),AKT4)   , (PARAM(81),RPS1)  ,
     1   (PARAM(82),RPS2)    , (PARAM(83),RPS3)   , (PARAM(84),RPS4)   ,
     1   (PARAM(85),B1)      , (PARAM(86),B2)     , (PARAM(87),B3)
      EQUIVALENCE
     1                         (PARAM( 89),DEL1DN),(PARAM( 90),DEL2DN),
     1   (PARAM( 91),DEL3DN),(PARAM( 92),DEL1IN), (PARAM( 93),DEL2IN),
     1   (PARAM( 94),DEL3IN), (PARAM( 95),PHIDN), (PARAM(96),PHIRN),
     1   (PARAM( 97),DFW1IN), (PARAM( 98),DFW2IN),(PARAM( 99),U1PIN),
     1   (PARAM(100),U2PIN),  (PARAM(101),U3PIN), (PARAM(102),U4PIN),
     1   (PARAM(103),S1PIN),  (PARAM(104),S2PIN), (PARAM(105),S3PIN),
     1   (PARAM(106),S4PIN), (PARAM(107),PPRT)
     1,(PARAM(110),TQMAX),(PARAM(111),AKTQ),(PARAM(112),VCIN)
     1,(PARAM(113),SWMT),(PARAM(114),DSWCM),(PARAM(115),TST),
     1(PARAM(116),DSLP),(PARAM(117),CGAM),(PARAM(118),CS)
     1  ,(PARAM(119),TQRBR),(PARAM(120),TQFBR)
     1  ,(PARAM(121),PFL),(PARAM(122),TTD),(PARAM(123),DSW)
     1   ,(PARAM(124),TSW)
      EQUIVALENCE
     1  (PARAM(130),AMCR),(PARAM(131),FSP),(PARAM(132),AKSL1),
     1  (PARAM(133),AKSL2),(PARAM(134),AA1),(PARAM(135),AA2),
     1  (PARAM(136),CCR),(PARAM(137),CFCR),(PARAM(138),AP),
     1  (PARAM(139),EP1),(PARAM(140),EP2),(PARAM(141),ERR1),
     1  (PARAM(142),ERR2),
     1  (PARAM(143),AML1),(PARAM(144),AML2),(PARAM(145),RRIM),
     1   (PARAM(146),RWR),(PARAM(147),AKV),
     1    (PARAM(148),AK1 ) , (PARAM(149),AK2 ) , (PARAM(150),AM   ) ,
     1    (PARAM(151),PM   ) , (PARAM(152),PV   ) , (PARAM(153),RT   ) ,
     1    (PARAM(154),TD1 ) , (PARAM(155),TD2 ) , (PARAM(156),TD3 ) ,
     1    (PARAM(157),TD4 ) , (PARAM(158),Y1   ) , (PARAM(159),YMAX) ,
     1    (PARAM(160),ZET1) , (PARAM(161),ZET2) , (PARAM(162),EX1 ) ,
     1    (PARAM(163),TBF ) , (PARAM(164),TBR ) , (PARAM(165),T1   ) ,
     1    (PARAM(166),WN1 ) , (PARAM(167),WN2 ) , (PARAM(168),R1   ) ,
     1    (PARAM(169),PA0 )
     1   ,(PARAM(196),EPSK1),(PARAM(197),EPSK2)
     1  ,(PARAM(274),TLF),(PARAM(275),ZR0),(PARAM(276),ALF)
      EQUIVALENCE
     1   (PARAM(281),TSF),(PARAM(282),TSR),(PARAM(283),TANP),
     1   (PARAM(284),HFC),(PARAM(285),HRC),(PARAM(286),CRRC)
C#######################################
C### PARAM(190)-(195) ADDED 8/11/72 ###
C#######################################
      EQUIVALENCE
     1   (PARAM(290),ROT),(PARAM(291),RA0),(PARAM(292),RA1),
     1   (PARAM(293),RA2),(PARAM(294),RA3),(PARAM(295),RA4)
      REAL*4       IOUT   ,IN     ,ITMP
      INTEGER*4    OSECB        ,DONECB
      INTEGER*2    RTSW   ,RBSW   ,LDTSW ,OPEN   ,OPDN
C     CALL HYTST(3,0)                                                   00003620
C     CALL ADDR(IADC,INA)                                               00003630
      DO 10 I=1,48                                                      00003640
C     CALL ADDR(IDAC(I),IOUTA(I))                                       00003650
      IOUT(I)=0                                                         00003660
   10 CONTINUE                                                          00003670
      N1=295
C     N2=155
      N2=119
         RTSW    = 0
         RBSW    = 0
         OPEN    = 0
      CONSL = 1
      KEYBD = 5
      ITTY  = 6
      IPTR  = 3
      IPTP  = 4
      ICDRD  = 1
      LPTP   =2
      ICDPN  =7
      CALL TYPER(KEYBD,ITTY,1,0,0)
      WRITE(ITTY,11)
      WRITE(ITTY,10070)
      WRITE(ITTY,10120)
      READ(KEYBD,11000)LL
      CALL SACN(1,ISACNE)
      CALL SAMO(1,ISAMOE)
      CALL SLMO(3,ISLMOE)
      CALL SLMO(1,ISLMOE)
11000 FORMAT(I1)
10120 FORMAT(1H ,18HTYPE CR WHEN READY)
10070 FORMAT(1H0,27HENGAGE PATCH PANEL FOR TEST)
   11 FORMAT(T10,'VEHICLE HANDLING SIMULATION')
 8888 CONTINUE                                                          00003680
      XPF=XIN                                                           00003690
      YSPEC=YIN                                                         00003700
      NRUN=1                                                            00003710
 3333 FORMAT(20A4)                                                      00003720
      DO 6140 I=1,48                                                    00003730
      ITMP(I)=0                                                         00003740
 6140 CONTINUE                                                          00003750
      READ(ICDRD,3333)(ALPH(I),I=1,20)                                  00003760
      READ(ICDRD, 900)(PHIFNT(I),I=1,7)                                 00003770
      READ(ICDRD, 900)(THEFNT(I),I=1,7)                                 00003780
      READ(ICDRD,900) (PSIFNT(I),I=1,7)                                 00003790
      READ(ICDRD,900)(PHIRR(I),I=1,7)                                   00003800
      READ(ICDRD,900) (THERR(I),I=1,7)                                  00003810
      READ(ICDRD,900)(PSIRR(I),I=1,7)                                   00003820
      NTF=1                                                             00003830
  200 READ(ICDRD,900)   PBF(NTF),TQBF(NTF)                              00003840
      IF(PBF(NTF).GE.99999.0)GO TO 210                                  00003850
      NTF=NTF+1                                                         00003860
      GO TO 200                                                         00003870
  210 NTF=NTF-1                                                         00003880
      NTR=1                                                             00003890
  220 READ(ICDRD,900) PBR(NTR),TQBR(NTR)                                00003900
```

```
      IF(PBR(NTR).GE.99999.0)GO TO 230
      NTR=NTR+1
      GO TO 220
  230 NTR=NTR-1
      NMA=1
  240 READ(ICDRD,900)AFRI(NMA),AMA(NMA)
      IF(AFRI(NMA).GE.99999.0)GO TO 250
      NMA=NMA+1
      GO TO 240
  250  NMA=NMA-1
      NMB=1
  260 READ(ICDRD,900)GAMM(NMB)  ,AMB(NMB)
      IF(GAMM(NMB).GE.99999.0)GO TO 270
      NMB=NMB+1
      GO TO 260
  270 NMB=NMB-1
      NFA=1
  280 READ(ICDRD,900)GAMF(NFA),AFA(NFA)
      IF(GAMF(NFA).GE.99999.0)GO TO 290
      NFA=NFA+1
      GO TO 280
  290 NFA=NFA-1
      NFB=1
  300 READ(ICDRD,900)SLPP(NFB),AFB(NFB)
      IF(SLPP(NFB).GE.99999.0)GO TO 310
      NFB=NFB+1
      GO TO 300
  310 NFB=NFB-1
      INT=1
  900 FORMAT(8E10.0)
C 500 CALL INITA(IERR,0)
C     IF(IERR.NE.0)GO TO 500
C     CALL HOOKUP(0,0)
 2000 CALL SBPG7
      IF(PARAM(280).EQ.0.0)GO TO 3000
      GO TO 8888
 3000 CALL SBPG4
      ISTP=PARAM(109)+0.5
      IF(ISTP.NE.0)GO TO 2000
      CALL WAITBU (200)
         CALL RTMON
C     CALL WAIT(20000)
C     CALL INMUX(IERR,1)
C4000 CALL OP(IERR)
C     CALL RUN(IERR)
  999 FORMAT('0' JUST CALLED RUN-OP')
C5000 WAIT
 2468 CONTINUE
C     LI,7   X'FF00'
C     WD,7   X'1102'
      DO 3210 I=1,48
      IF(I.EQ.19.OR.I.EQ.20)ITMP(I)=IOUT(I)
C     CALL STIND(IERR,IDAC(I),ITMP(I))
 3210 CONTINUE
      CALL LBDAEP(00,39,ITMP,ILBDAE)
      CALL TLDA
      TEMP=TIME10-TIME25
      XXX=DOUT/12.0+3.67*(TIME5-TIME10)
      EFF=0.0
      IF(ADOUT.NE.0.0)EFF=ANOUT/ADOUT
```

```
C     PRINT 2345,TEMP,RMAX,ETAMAX,PSI5,PHIMAX,XXX,TOO,EFF                00004510
      WRITE(ITTY,2345)TEMP,RMAX,ETAMAX,PSI5,PHIMAX,XXX,TOO,EFF           00004510
 2345 FORMAT('0 TIME INTERVAL=',F15.5,', RMAX=',F15.5,', ETAMAX=',F15.5/
     1   '0PSI AT T=5 =',F15.5,',PHIMAX=',F15.5,', D=',F15.5,', X=',F15.
     25, ', EFF=',F15.5)
C2345 FORMAT('0 TIME INTERVAL='F15.5,', RMAX='F15.5,', ETAMAX='F15.5/   00004520
C    1   '0PSI AT T=5 ='F15.5,', PHIMAX='F15.5,', D='F15.5,', X='F15.5, 00004530
C    1   ', EFF='F15.5)                                                  00004540
      IF(PARAM(189).EQ.0.0)GO TO 2469                                    00004550
      IRUNTB(NRUN)=PARAM(044)+0.5                                        00004560
      VTB(NRUN)=DSWMAX                                                   00004570
      BTVTB(NRUN)=BTVMAX*57.3                                            00004580
      SNRTB(NRUN)=TQFMAX/12.0                                            00004590
      SNLTB(NRUN)=TQRMAX/12.0                                            00004600
      TMPTB(NRUN)=EFF                                                    00004610
      DSWTB(NRUN)=TEMP                                                   00004620
      BTTB(NRUN)=ETAMAX                                                  00004630
      DTB(NRUN)=RMAX                                                     00004640
      EFFTB(NRUN)=TMAX1                                                  00004650
      PSITB(NRUN)=TMAX2                                                  00004660
      RTAB(NRUN)=TMAX3                                                   00004670
      DLYTB(NRUN)=PSIOUT*57.3                                            00004680
      EXTAB(NRUN)=PHIMAX                                                 00004690
      IF(PARAM(188).NE.0.0)GO TO 8765                                    00004700
      NRUN=NRUN+1                                                        00004710
      GO TO 2469                                                         00004720
C8765 PRINT 5984,(ALPH(I),I=1,20)                                         00004730
 8765 WRITE( LPTR,5984 ) ( ALPH(I),I=1,20 )                              00004730
 5984 FORMAT('1',20A4)                                                   00004740
C     PRINT 988                                                          00004750
      WRITE(ITTY,988)                                                    00004750
      DO 8764 I=1.NRUN                                                   00004760
C     PRINT 987.IRUNTB(I),VTB(I),SNRTB(I),SNLTB(I),TMPTB(I),DSWTB(I),    00004770
C    1   BTTB(I),                                                        00004780
      WRITE(ITTY,987) IRUNTB(I),VTB(I),SNRTB(I),SNLTB(I),TMPTB(I),       00004770
     1DSWTB(I),BTTB(I),                                                  00004780
     1     DTB(I),EFFTB(I), PSITB(I),RTAB(I),DLYTB(I),EXTAB(I),BTVTB(I)  00004790
 8764 CONTINUE                                                           00004800
  988 FORMAT('0 RUN',6X,'DSW',5X,'TQBF',5X,'TQBR',3X,'EBRAKE',3X,'TBRAKE00004810
     1',4X,'AYMAX',5X,'RMAX',5X,'RBAR',5X,'RHAT',5X,'DSW*',3X,'DELPSI',  00004820
     13X,'DELPHI',3X,'BTVMAX')                                           00004830
  987 FORMAT(I5.3F9.0,F9.3,F9.2,F9.3,F9.1,2F9.3,F9.2,F9.0,2F9.1)         00004840
      NRUN=1                                                             00004850
      XPF=XIN                                                            00004860
      YSPEC=YIN                                                          00004870
C2469 IF(ISW7.EQ.1)GO TO 3000                                             00004880
C     GO TO 2000                                                         00004890
 2469 CONTINUE
C     CONTINUATION DIRECTIVE
  400 WRITE(ITTY,10140)
10140 FORMAT(1H ,' TYPE CONTINUATION DIRECTIVE- 1-TERMINATE, 2-REPEAT, 3
     1-RESTART')
  510    READ( KEYBD,11000 ) ISW7
      GO TO ( 440,3000,2000 ),ISW7
      WRITE(ITTY,12020)
12020 FORMAT(1H ,32H**ILLEGAL CONTINUATION DIRECTIVE)
      GO TO 510
12030 FORMAT(1H ,13HCO COMPONENT ,A2,1X,I3,10H WRONG BY ,I6,11H MILLIVOL
     1TS)
99999 FORMAT(1H ,52HCONSOLE FAILED TO SELECT- RETURN TO KEYBOARD MONITOR
     1)
  440 CONTINUE
      IF( RTSW.EQ.0 ) GO TO 441
      OSECB    =0
         CALL HPOST( DONECB,'J007' )
      CALL WAITRT(OSECB)
  441 CONTINUE
C
      WRITE(ITTY,10150)
10150 FORMAT(1X,10HTERMINATED)
C
      CALL RACN(1,IRACNE)
      CALL CHKIO
      CALL WRTOFF
      CALL RDOFF
      STOP
      END
```

```
C     SUBROUTINE POTSET                                                 00006579
      SUBROUTINE POTSET                                                 00006580
      DIMENSION SS(4,4),BP(4,2),ALBP(4,2),SSI(4,4)                      00006590
      COMMON/DEVICE/KEYBD,ITTY,ICDRD,LPTR
      COMMON/HHHH/H1,H2,H3,H4                                           00006600
      COMMON/SPLTAX/SPSR3,CPSR3,SPSR4,CPSR4,SCR3,SCR4,TBCR3,TBCR4,TBSR3,00006610
     1   TBSR4,TRSR3,TRSR4,TRCR3,TRCR4,PSR3,PSR4,IAX                    00006620
      COMMON/NUTS/TOO                                                   00006630
      COMMON/CLEAN/ONEOA,ONEOD,PBSW,PB                                  00006640
      COMMON/VARS/P,Q,R,U,V,W,X,Y,Z,THE,PHI,PSI,PO,QO,RO,UO,VO,WO,XO,   00006650
     1     YO,ZO.THEO,PHIO,PSIO                                         00006660
      COMMON/ZILCH/TQMAXP,AKTQP                                         00006670
      COMMON/ANALOG/DEL1DT.DEL2DT,DEL3DT,DEL1,DEL2,DEL3,PHIRD,PHIR,     00006680
     1     DELFW1,DELFW2,U1P,U2P,U3P,U4P,S1P,S2P,S3P,S4P,TTA,PFSLP      00006690
      COMMON/INOUT/INA(32),IOUTA(48),IN(32),IOUT(48),ISW1,ISW7,SFIN(32),00006700
     1      SFOUT(48),IPRT,ITMP(48)                                     00006710
      COMMON/EFFS/ANUM,ADEN,ANUMDT,ADENDT,ANUMO,ADENO,ANUMDO,ADENDO,    00006720
     1   ANOUT,ADOUT                                                    00006730
      COMMON/COMBLK/AIXP,SM,AIYP,AIXZP,GAM1,GAM2,GAM3,AIXBR,AIYBR,      00006740
     1     AIZBR.A1.A2,AIXZBR,A12,E1,E2,E3,DELTA,GV1,GV2,GP1,GP2,GR1,   00006750
     1     GR2,CIP,CIVP,RZF,RZR,A2T,CA20,CA23,TIME,DT,      ANGNL,ANGNLO  00006760
     1     ,TRO2.TFO2,TSO2,G,THRD,TWN7                                  00006770
      COMMON/FRIDAY/BTVMAX,BTVTB(100)                                   00006780
      COMMON/UVW/VC,UIN                                                 00006790
      COMMON/SP7BLK/N1,N2,IPOT(244),IPOTAD(244).PARAM(300)              00006800
      COMMON/MICKEY/IRUNTB(100),VTB(100),SNLTB(100),SNRTB(100),DSWTB(10000006810
     1  ),BTTB(100),DTB(100),EFFTB(100),DYTB(100),PSITB (100),NRUN,     00006820
     1   YSPEC,PSIM,XPF                                                 00006830
     1   ,TMPTB(100)                                                    00006840
      EQUIVALENCE                                                       00006850
     1   (PARAM( 1),AMS)     , (PARAM( 2),AMUF)  , (PARAM( 3),AMUR)   , 00006860
     1   (PARAM( 4),ZF)      , (PARAM( 5),ZR)    , (PARAM( 6),A)      , 00006870
     1   (PARAM( 7),B)       , (PARAM( 8),TF)    , (PARAM( 9),TR)     , 00006880
     1   (PARAM(10),TS)      , (PARAM(11),AIX)   , (PARAM(12),AIY)    , 00006890
     1   (PARAM(13),AIZ)     , (PARAM(14),AIXZ)  , (PARAM(15),AIR)    , 00006900
     1   (PARAM(16),CF)      , (PARAM(17),RF)    , (PARAM(18),CFP)    , 00006910
     1   (PARAM(19),AKF)     , (PARAM(20),ALAMF) , (PARAM(21),OFC)    , 00006920
     1   (PARAM(22),OFT)     , (PARAM(23),CR)    , (PARAM(24),RR)     , 00006930
     1   (PARAM(25),CRP)     , (PARAM(26),AKR)   , (PARAM(27),ALAMR)  , 00006940
     1   (PARAM(28),ORC)     , (PARAM(29),ORT)   , (PARAM(30),AKRS)   , 00006950
     1   (PARAM(31),RW)      , (PARAM(32),B4)    , (PARAM(33),OT)     , 00006960
     1   (PARAM(34),CA0)     , (PARAM(35),CA1)   , (PARAM(36),CA2)    , 00006970
     1   (PARAM(37),CA3)     , (PARAM(38),CA4)   , (PARAM(39),AISW)   , 00006980
     1   (PARAM(44),AKDL)    , (PARAM(41),AKSC)  , (PARAM(42),ANG)    , 00006990
     1   (PARAM(43),WG)      , (PARAM(40),ANL2)  , (PARAM(45),AKSL)     00007000
      EQUIVALENCE                                                       00007010
     1   (PARAM(46),ANL1)    , (PARAM(47),AIFW)  , (PARAM(48),HDL)    , 00007020
     1   (PARAM(49),AIWF)    , (PARAM(50),AIWR)  , (PARAM(51),AID)    , 00007030
     1   (PARAM(52),ARBR)    , (PARAM(53),EPS1)  , (PARAM(54),EPS2)   , 00007040
     1   (PARAM(55),PTBR)    , (PARAM(56),YSA1)  , (PARAM(57),YSA2)   , 00007050
     1   (PARAM(58),YHS1)    , (PARAM(59),YHS2)  , (PARAM(60),AKD)    , 00007060
     1   (PARAM(61),TQDBR)   , (PARAM(62),AK)    , (PARAM(63),PIN)    , 00007070
     1   (PARAM(64),QIN)     , (PARAM(65),RIN)   , (PARAM(66),UIZ)    , 00007080
     1   (PARAM(67),VIN)     , (PARAM(68),WIN)   , (PARAM(69),XIN)    , 00007090
     1   (PARAM(70),YIN)     , (PARAM(71),ZIN)   , (PARAM(72),THEIN)  , 00007100
     1   (PARAM(73),PHIIN)   , (PARAM(74),PSIIN) , (PARAM(75),DTIN)   , 00007110
     1   (PARAM(76),TEND)    , (PARAM(77),AKT1)  , (PARAM(78),AKT2)   , 00007120
     1   (PARAM(79),AKT3)    , (PARAM(80),AKT4)  , (PARAM(81),RPS1)   , 00007130
     1   (PARAM(82),RPS2)    , (PARAM(83),RPS3)  , (PARAM(84),RPS4)   , 00007140
     1    (PARAM(85),B1)      , (PARAM(86),B2)      , (PARAM(87),B3)      00007150
      EQUIVALENCE                                                       00007160
     1                          (PARAM( 89),DEL1DN),(PARAM( 90),DEL2DN), 00007170
     1    (PARAM( 91),DEL3DN),(PARAM( 92),DEL1IN), (PARAM( 93),DEL2IN), 00007180
     1    (PARAM( 94),DEL3IN), (PARAM( 95),PHIDN), (PARAM(96),PHIRN),   00007190
     1    (PARAM( 97),DFW1IN), (PARAM( 98),DFW2IN),(PARAM( 99),U1PIN),  00007200
     1    (PARAM(100),U2PIN),  (PARAM(101),U3PIN), (PARAM(102),U4PIN),  00007210
     1    (PARAM(103),S1PIN),  (PARAM(104),S2PIN), (PARAM(105),S3PIN),  00007220
     1    (PARAM(106),S4PIN), (PARAM(107),PPRT)                         00007230
     1,(PARAM(110),TQMAX),(PARAM(111),AKTQ),(PARAM(112),VCIN)           00007240
     1,(PARAM(113),SWMT),(PARAM(114),DSWCM),(PARAM(115),TST),           00007250
     1(PARAM(116),DSLP),(PARAM(117),CGAM),(PARAM(118),CS)               00007260
     1  ,(PARAM(119),TQRBR),(PARAM(120),TQFBR)                          00007270
     1  ,(PARAM(121),PFL),(PARAM(122),TTD),(PARAM(123),DSW)             00007280
     1    ,(PARAM(124),TSW)                                             00007290
      EQUIVALENCE                                                       00007300
     1   (PARAM(130),AMCR),(PARAM(131),ESP),(PARAM(132),AKSL1),         00007310
     1   (PARAM(133),AKSL2),(PARAM(134),AA1),(PARAM(135),AA2),          00007320
     1   (PARAM(136),CCR),(PARAM(137),CFCR),(PARAM(138),AP),            00007330
     1   (PARAM(139),EP1),(PARAM(140),EP2),(PARAM(141),FRR1),           00007340
     1   (PARAM(142),ERR2),                                             00007350
     1   (PARAM(143),AML1),(PARAM(144),AML2),(PARAM(145),RRIM),         00007360
     1    (PARAM(146),RWR),(PARAM(147),AKV),                            00007370
     1     (PARAM(148),AK1 ) , (PARAM(149),AK2 ) , (PARAM(150),AM   ) , 00007380
     1     (PARAM(151),PM  ) , (PARAM(152),PV  ) , (PARAM(153),RT   ) , 00007390
     1     (PARAM(154),TD1 ) , (PARAM(155),TD2 ) , (PARAM(156),TD3 ) ,  00007400
     1     (PARAM(157),TD4 ) , (PARAM(158),Y1  ) , (PARAM(159),YMAX) ,  00007410
     1     (PARAM(160),ZET1) , (PARAM(161),ZET2) , (PARAM(162),EX1 ) ,  00007420
     1     (PARAM(163),THF ) , (PARAM(164),TBR ) , (PARAM(165),T1   ) , 00007430
     1     (PARAM(166),WN1 ) , (PARAM(167),WN2 ) , (PARAM(168),R1   ) , 00007440
     1     (PARAM(169),PA0 )                                            00007450
     1    ,(PARAM(196),EPSK1),(PARAM(197),EPSK2)                        00007460
      EQUIVALENCE                                                       00007470
     1    (PARAM(223),CR1C),(PARAM(224),CR1T),(PARAM(225),CR2C),        00007480
     1    (PARAM(226),CR2T),(PARAM(227),CR3C),(PARAM(228),CR3T),        00007490
     1    (PARAM(229),CR4C),(PARAM(230),CR4T),(PARAM(231),AH1),         00007500
     1    (PARAM(232),AH2),(PARAM(233),ANL),(PARAM(234),AKF1),          00007510
     1   (PARAM(235),AKF2),(PARAM(236),AKR3),(PARAM(237),AKR4)          00007520
      EQUIVALENCE (SS(1,1),PARAM(242)),(BP(1,1),PARAM(258)),(ALBP(1,1), 00007530
     1    PARAM(266))                                                   00007540
     1   ,(PARAM(274),TLF),(PARAM(275),ZR0),(PARAM(276),ALF)            00007550
      EQUIVALENCE                                                       00007560
     1    (PARAM(281),TSF),(PARAM(282),TSR),(PARAM(283),TANP),          00007570
     1    (PARAM(284),HFC),(PARAM(285),HRC),(PARAM(286),CRRC)           00007580
      EQUIVALENCE
     1    (PARAM(290),ROT),(PARAM(291),RA0),(PARAM(292),RA1),
     1    (PARAM(293),RA2),(PARAM(294),RA3),(PARAM(295),RA4)
C     DATA N1/289/,N2/155/,T/10000.0/,TIN/0.0001/                       00007590
      DATA                   T/10000.0/,TIN/0.0001/                     00007590
C     N1 , N2 EQUATED TO THEIR VALUES IN MAIN                               7591
      IAX=PARAM(287)+0.5                                                00007600
      PSR3=PARAM(288)*.0174533                                          00007610
      PSR4=PARAM(289)*.0174533                                          00007620
C     SPSR3=SIN(PSR3)                                                   0
C     SPSR4=SIN(PSR4)                                                   00007640
      SPSR3=(TAN(2.0*HFC/TF))*2.0/AMUF
      SPSR4=(TAN(2.0*HRC/TR))*2.0/AMUR
      CPSR3=COS(PSR3)                                                   00007650
      CPSR4=COS(PSR4)                                                   00007660
      SCR3=SPSR3*CPSR3                                                  00007670
      SCR4=SPSR4*CPSR4                                                  00007680
      TMP=2.0*B/TS                                                      00007690
      TBSP3=TMP*SPSR3                                                   00007700
      TBSP4=TMP*SPSR4                                                   00007710
      TBCP3=TMP*CPSR3                                                   00007720
      TBCP4=TMP*CPSR4                                                   00007730
      TMP=TR/TS                                                         00007740
      TRSP3=TMP*SPSR3                                                   00007750
      TRSP4=TMP*SPSR4                                                   00007760
      TRCP3=TMP*CPSR3                                                   00007770
      TRCP4=TMP*CPSR4                                                   00007780
C     DO 20 I=1,4                                                       00007790
C     SSI(I,1)=ALBP(I,1)-SS(I,1)*BP(I,1)                                00007800
C     SSI(I,2)=0.0                                                      00007810
C     SSI(I,3)=0.0                                                      00007820
C  20 SSI(I,4)=ALBP(I,2)-SS(I,4)*BP(I,2)                                00007830
      SM=AMS+AMUF+AMUR                                                  00007850
      UIN=UIZ*12.0*5280.0/3600.0                                        00007860
      VC=VCIN*12.0*5280.0/3600.0                                        00007870
      ISW7=1                                                            00007880
      IPRT=PARAM(108)+0.5                                               00007890
      TWO3=2.0/3.0                                                      00007900
      TEN=0.1                                                           00007910
      C1=2000.0                                                         00007920
      C2=60.0                                                           00007930
      C3=1000.0                                                         00007940
      HUN=0.01                                                          00007950
      TOU=0.001                                                         00007960
      TTO=0.0001                                                        00007970
      G=386.4                                                           00007980
      THRD=1.0/3.0                                                      00007990
      TWN7=1.0/27.0                                                     00008000
      H1=RW-(AMUF+B*AMS/(A+B))*G/(2.*AKT1)                              00008010
      H2=RW-(AMUF+B*AMS/(A+B))*G/(2.*AKT2)                              00008020
      H3=RW-(AMUR+A*AMS/(A+B))*G/(2.*AKT3)                              00008030
      H4=RW-(AMUR+A*AMS/(A+B))*G/(2.*AKT4)                              00008040
      ZIN=(B*(H1+ZF)+A*(H3+ZR))/(A+B)*(-1.)
      THETN=(H1-H3+ZF-ZR)/(A+B)*57.29578                                00008070
      AIBR=AIWR+AID*ARBR**2*0.25                                        00008080
      AIBRP=AIBR-AIWR                                                   00008090
      TOO=0.0                                                           00008100
      RPS1=UIN/H1                                                       00008110
      RPS2=UIN/H2                                                       00008120
      RPS3=UIN/H3                                                       00008130
      RPS4=UIN/H4                                                       00008140
      TFO2=TF*0.5                                                       00008150
      CRT=C*RT                                                          00008160
      TSO2=TS/2.0                                                       00008170
      TRO2=TR*0.5                                                       00008180
      TEMR=ALAMR*AKR-AKR                                                00008190
      TEMF=ALAMF*AKF-AKF                                                00008200
      AIXP=AMUF*ZF*ZF+AMUR*ZR*ZR                                        00008210
      AIYP=AIXP                                                         00008220
      AIZP=AMUF*(A*A+TFO2**2)+AMUR*B*B                                  00008230
      IF(IAX.EQ.1) GO TO 30
      AIZP=AIZP+AMUR*TRO2**2
   30 AIXZP=AMUF*A*ZF-AMUR*B*ZR                                         00008240
      GAM1=AMUF*A-AMUR*B                                                00008250
      GAM2=AMUF*ZF+AMUR*ZR                                              00008260
      GAM3=GAM2                                                         00008270
      AIXBR=AIX+AIXP                                                    00008280
      AIYBR=AIY+AIYP                                                    00008290
      AIZBR=AIZP +AIZ                                                   00008300
      A1=GAM2/SM                                                        00008310
      A2=AIYBR/GAM2                                                     00008320
      AIXZBR=AIXZP+AIXZ                                                 00008330
      CA23=CA2*CA3                                                      00008340
      A12=A1-A2                                                         00008350
      F1=AIXBR*AIZBR-AIXZBR**2                                          00008360
      F2=GAM1*AIXZBR-GAM3*AIZBR                                         00008370
      F3=GAM3*AIXZBR-GAM1*AIXBR                                         00008380
      DELTA=F1*SM+GAM2*E2+GAM1*E3                                       00008390
      GV1=GAM2*AIZBR-GAM1*AIXZBR                                        00008400
      GV2=GAM2*AIXZBR-GAM1*AIXBR                                        00008410
      GP1=SM*AIZBR-GAM1**2                                              00008420
      GP2=SM*AIXZBR-GAM1*GAM3                                           00008430
      GR1=GP2                                                           00008440
      GR2=SM*AIXBR-GAM2*GAM3                                            00008450
      CIP=B*AMS*G/(AMUF*(A+B))+G                                        00008460
      TQMAXP=TQMAX*ARBR*0.5                                             00008470
      AKTOP=AKTO*ARBR*0.5                                               00008480
      CIVP=A*AMS*G/(AMUR*(A+B))+G                                       00008490
      RZF=RW+ZF                                                         00008500
      RZR=RW+ZR                                                         00008510
      A2T= DT*CA2                                                       00008520
      CA20=CA0*CA2                                                      00008530
      ONEOA=1.0/A12                                                     00008540
      ONEOD=1.0/DELTA                                                   00008550
      PBSW=0.0                                                          00008560
      PB=0.0                                                            00008570
C     IPOT(3)=10000                                                     00008580
C     IPOT(5)=10000                                                     00008590
C     IPOT(6)=1.0/(10.0*AMCR)*T+0.5                                     00008600
C     IPOT(7)=ESP/(2.0*10.0*57.3)*T+0.5                                 00008610
C     IPOT(8)=ESP/(2.0*10.0*57.3)*T+0.5                                 00008620
C     IPOT(11)=RF/(TF*TF*100.0)*T+0.5                                   00008630
      IPOT(08)=RF/(TF*TF*100.0)*T                                       00008630
C     IPOT(12)=RF/(TF*TF*100.0)*T+0.5                                   00008640
      IPOT(14)=RF/(TF*TF*100.0)*T                                       00008640
C     IPOT(13)=AKSL1/(AA1*40000.0)*T+0.5                                00008650
C     IPOT(14)=AKSL2/(AA2*40000.0)*T+0.5                                00008660
C     IPOT(15)=CCR/100.0*T+0.5                                          00008670
C     IPOT(16)=AKSC*ANG/(AP*2000.0)*T+0.5                               00008680
C     IPOT(17)=CFCR/20000.0*T+0.5                                       00008690
C     IPOT(19)=3.0*ANG/(10.0*AP)*T+0.5                                  00008700
C     IPOT(20)=6.0/AA2*T+0.5                                            00008710
C     IPOT(21)=2.0*EP2/(2.0*57.3)*T+0.5                                 00008720
C     IPOT(22)=2.0*EP2/(2.0*57.3)*T+0.5                                 00008730
C     IPOT(23)=2.0*EP1/(2.0*57.3)*T+0.5                                 00008740
C     IPOT(24)=2.0*EP1/(2.0*57.3)*T+0.5                                 00008750
C     IPOT(25)=2.0/3.0*T+0.5                                            00008760
C     IPOT(27)=AKSL1/(20000.0*AIFW)*T+0.5                               00008770
C     IPOT(28)=AKSL2/(20000.0*AIFW)*T+0.5                               00008780
C     IPOT(29)=AH1/(100.0*AIFW)*T+0.5                                   00008790
C     IPOT(30)=AH2/(100.0*AIFW)*T+0.5                                   00008800
C     IPOT(33)=FRR1/20.0*T+0.5                                          00008810
C     IPOT(34)=FRR1/20.0*T+0.5                                          00008820
C     IPOT(35)=6.0/AA1*T+0.5                                            00008830
C     IPOT(36)=SS(2,3)/10.0*T+0.5                                       00008840
C     IPOT(037)=(2.0*AKT1/(5000.0*AMUF))*T+0.5                          00008850
```

```
      IPOT(32)=(2.0*AKT1/(5000.0*AMUF))*T                               0008850
C     IPOT(41)=ERR2/20.0*T+0.5                                          00008870
C     IPOT(42)=ERR2/20.0*T+0.5                                          00008880
C     IPOT(43)=SS(4,2)/10.0*T+0.5                                       00008890
C     IPOT(44)=SS(4,3)/10.0*T+0.5                                       00008900
C     IPOT(55)=SS(1,2)/10.0*T+0.5                                       00008910
C     IPOT(56)=SS(1,2)/10.0*T+0.5                                       00008920
C     IPOT(57)=SS(1,1)/10.0*T+0.5                                       00008930
C     IF(SSI(1,1).LT.0) SSI(1,1)=-SSI(1,1)                              00008940
C     IPOT(58)=SSI(1,1)/1000.0*T+0.5                                    00008950
C     IPOT(59)=SS(1,3)/10.0*T+0.5                                       00008960
C     IPOT(60)=SS(1,3)/10.0*T+0.5                                       00008970
C     IPOT(61)=RP(1,1)/100.0*T+0.5                                      00008980
C     IPOT(61)=-IPOT(61)                                                00008990
C     IPOT(62)=RP(1,2)/100.0*T+0.5                                      00009000
C     IPOT(63)=SS(1,4)/10.0*T+0.5                                       00009010
C     IF(SSI(1,4).LT.0) SSI(1,4)=-SSI(1,4)                              00009020
C     IPOT(64)=SSI(1,4)/1000.0*T+0.5                                    00009030
C     IPOT(65)=SS(2,2)/10.0*T+0.5                                       00009040
C     IPOT(66)=SS(2,2)/10.0*T+0.5                                       00009050
C     IPOT(67)=SS(2,1)/10.0*T+0.5                                       00009060
C     IF(SSI(2,1).LT.0) SSI(2,1)=-SSI(2,1)                              00009070
C     IPOT(68)=SSI(2,1)/1000.0*T+0.5                                    00009080
C     IPOT(69)=RP(2,1)/100.0*T+0.5                                      00009090
C     IPOT(69)=-IPOT(69)                                                00009100
C     IPOT(038)=(2.0*AKT2/(5000.0*AMUF))*T+0.5                          00008860
      IPOT(05)=(2.0*AKT2/(5000.0*AMUF))*T                               0008860
C     IPOT(70)=RP(2,2)/100.0*T+0.5                                      00009110
C     IPOT(72)=SS(2,3)/10.0*T+0.5                                       00009120
C     IPOT(73)=SS(2,4)/10.0*T+0.5                                       00009130
C     IF(SSI(2,4).LT.0) SSI(2,4)=-SSI(2,4)                              00009140
C     IPOT(74)=SSI(2,4)/1000.0*T+0.5                                    00009150
C     IPOT(75)=SS(3,2)/10.0*T+0.5                                       00009160
C     IPOT(76)=SS(3,2)/10.0*T+0.5                                       00009170
C     IPOT(77)=RP(3,1)/100.0*T+0.5                                      00009180
C     IPOT(77)=-IPOT(77)                                                00009190
C     IPOT(78)=RP(3,2)/100.0*T+0.5                                      00009200
C     IPOT(79)=SS(3,1)/10.0*T+0.5                                       00009210
C     IF(SSI(3,1).LT.0) SSI(3,1)=-SSI(3,1)                              00009220
C     IPOT(80)=SSI(3,1)/1000.0*T+0.5                                    00009230
C     IF(IPOT(80).LT.0) IPOT(80)=-IPOT(80)                              00009240
C     IPOT(81)=SS(3,3)/10.0*T+0.5                                       00009250
C     IPOT(82)=SS(3,3)/10.0*T+0.5                                       00009260
C     IPOT(83)=SS(3,4)/10.0*T+0.5                                       00009270
C     IF(SSI(3,4).LT.0) SSI(3,4)=-SSI(3,4)                              00009280
C     IPOT(84)=SSI(3,4)/1000.0*T+0.5                                    00009290
C     IPOT(85)=RP(4,1)/100.0*T+0.5                                      00009300
C     IPOT(85)=-IPOT(85)                                                00009310
C     IPOT(86)=RP(4,2)/100.0*T+0.5                                      00009320
      IPOT(086)=0.2000*T
C     IPOT(87)=SS(4,2)/10.0*T+0.5                                       00009330
C     IPOT(88)=SS(4,1)/10.0*T+0.5                                       00009340
C     IF(SSI(4,1).LT.0) SSI(4,1)=-SSI(4,1)                              00009350
C     IPOT(89)=SSI(4,1)/1000.0*T+0.5                                    00009360
C     IF(IPOT(89).LT.0) IPOT(89)=-IPOT(89)                              00009370
C     IPOT(90)=SS(4,3)/10.0*T+0.5                                       00009380
C     IPOT(91)=SS(4,4)/10.0*T+0.5                                       00009390
C     IF(SSI(4,4).LT.0) SSI(4,4)=-SSI(4,4)                              00009400
C     IPOT(92)=SSI(4,4)/1000.0*T+0.5                                    00009410
C     IPOT(96)=AKF1*10.0+0.5                                            00009420
```

```
      IPOT(096)=0.2000*T
C     IPOT(97)=(ALAMF*AKF1-AKF1)*10.0+0.5                               00009430
C     IPOT(98)=OFC*(ALAMF*AKF1-AKF1)*1.0+0.5                            00009440
C     IPOT(98)=-IPOT(98)                                                00009450
C     IPOT(99)=IPOT(97)                                                 00009460
C     IPOT(100)=OFT*(ALAMF*AKF1-AKF1)*1.0+0.5                           00009470
C     IPOT(104)=AKF2*10.0+0.5                                           00009480
C     IPOT(105)=(ALAMF*AKF2-AKF2)*10.0+0.5                              00009490
C     IPOT(106)=OFC*(ALAMF*AKF2-AKF2)*1.0+0.5                           00009500
C     IPOT(106)=-IPOT(106)                                              00009510
C     IPOT(107)=IPOT(105)                                               00009520
C     IPOT(108)=OFT*(ALAMF*AKF2-AKF2)*1.0+0.5                           00009530
C     IPOT(112)=AKR3*10.0+0.5                                           00009540
C     IPOT(113)=(ALAMR*AKR3-AKR3)*10.0+0.5                              00009550
C     IPOT(114)=ORC*(ALAMR*AKR3-AKR3)*1.0+0.5                           00009560
C     IPOT(114)=-IPOT(114)                                              00009570
C     IPOT(115)=IPOT(113)                                               00009580
C     IPOT(116)=ORT*(ALAMR*AKR3-AKR3)*1.0+0.5                           00009590
C     IPOT(120)=AKR4*10.0+0.5                                           00009600
C     IPOT(121)=(ALAMR*AKR4-AKR4)*10.0+0.5                              00009610
C     IPOT(122)=ORC*(ALAMR*AKR4-AKR4)*1.0+0.5                           00009620
C     IPOT(122)=-IPOT(122)                                              00009630
C     IPOT(123)=IPOT(121)                                               00009640
C     IPOT(124)=ORT*(ALAMR*AKR4-AKR4)*1.0+0.5                           00009650
      IPOT(92)=57.3/200.*T                                              00009653
      IPOT(97)=IPOT(92)                                                 00009654
      IPOT(100)=2.*IPOT(92)                                             00009655
      IPOT(105)=2.*IPOT(97)                                             00009656
C     IPOT(125)=HUN*RPS1*T+0.5                                          00009660
      IPOT(101)=HUN*RPS1*T                                              00009660
C     IPOT(126)=HUN*RPS2*T+0.5                                          00009670
      IPOT(104)=HUN*RPS2*T                                              00009670
C     IPOT(127)=(4.0/AIWF)*T*PARAM(239)+0.5                             00009680
      IPOT(103)=(4.0/AIWF)*T*PARAM(239)                                 00009680
C     IPOT(128)=(60.0*AKT2/(10000.0*AIWF))*T+0.5                        00009690
      IPOT(102)=(60.0*AKT2/(10000.0*AIWF))*T                            00009690
C     IPOT(129)=(4.0/AIWF)*T*PARAM(238)+0.5                             00009700
      IPOT(52 )=(4.0/AIWF)*T*PARAM(238)                                 00009700
C     IPOT(130)=(60.0*AKT1/(10000.0*AIWF))*T+0.5                        00009710
      IPOT(51 )=(60.0*AKT1/(10000.0*AIWF))*T                            00009710
C     IPOT(131)=1.5*T+0.5                                               00009720
C     IPOT(132)=IPOT(131)                                               00009730
C     IPOT(133)=TOU*2.0*CFP*T+0.5                                       00009740
      IPOT(19 )=.1*TOU*2.0*CFP*T                                        00009740
C     IPOT(134)=IPOT(133)                                               00009750
      IPOT(24 )=IPOT(19 )                                               00009750
C     IPOT(135)=0.1333*T+0.5                                            00009760
      IPOT(87 )=0.1333*T                                                00009760
C     IPOT(136)=(RW/15.0)*T+0.5                                         00009770
      IPOT(88 )=(RW/15.0)*T                                             00009770
C     IPOT(137)=IPOT(135)                                               00009780
      IPOT(99 )=IPOT(87 )                                               00009780
C     IPOT(138)=IPOT(136)                                               00009790
      IPOT(98 )=IPOT(88 )                                               00009790
C     IPOT(139)=1.5*T+0.5                                               00009800
C     IPOT(140)=1.5*T+0.5                                               00009810
C     IPOT(141)=(2.0/(10.0*AMUF))*T+0.5                                 00009820
      IPOT(02 )=(2.0/(10.0*AMUF))*T                                     00009820
C     IPOT(142)=(HUN*RF/(TF*TF))*T+0.5                                  00009830
      IPOT(04 )=(HUN*RF/(TF*TF))*T                                      00009830
```

```
C     IPOT(144)=TOU*CFP*T+0.5                                           00009840
      IPOT(16 )=TOU*CFP*T                                               00009840
C     IPOT(147)=(0.2/AMS)*T+0.5                                         00009850
      IPOT(01 )=(0.1/AMS)*T                                             00009850
C     IPOT(148)=(0.2/AMS)*T+0.5                                         00009860
      IPOT(06 )=(0.1/AMS)*T                                             00009860
C     IPOT(149)=IPOT(141)                                               00009870
      IPOT(07 )=IPOT(02 )                                               00009870
C     IPOT(150)=IPOT(142)                                               00009880
      IPOT(03 )=IPOT(04 )                                               00009880
C     IPOT(152)=IPOT(144)                                               00009890
      IPOT(22 )=IPOT(16 )                                               00009890
C     IPOT(153)=HUN*RPS3*T+0.5                                          00009900
      IPOT(110)=HUN*RPS3*T                                              00009900
C     IPOT(154)=HUN*RPS4*T+0.5                                          00009910
      IPOT(114)=HUN*RPS4*T                                              00009910
C     IPOT(156)=AIBRP/AIBR*T+0.5                                        00009920
      IPOT(119)=AIBRP/AIBR*T                                            00009920
C     IPOT(157)=(4.0/AIWR)*T*PARAM(240)+0.5                             00009930
      IPOT(112)=(4.0/AIBR)*T*PARAM(240)                                 00009930
C     IPOT(158)=(60.0*AKT3/(10000.0*AIWR))*T+0.5                        00009940
      IPOT(113)=(60.0*AKT3/(10000.0*AIBR))*T                            00009940
C     IPOT(159)=1.5*T+0.5                                               00009950
C     IPOT(160)=1.5*T+0.5                                               00009960
C     IPOT(164)=IPOT(156)                                               00009970
      IPOT(50 )=IPOT(119)                                               00009970
C     IPOT(165)=(4.0/AIWR)*T*PARAM(241)+0.5                             00009980
      IPOT(111)=(4.0/AIBR)*T*PARAM(241)                                 00009980
C     IPOT(166)=(60.0*AKT4/(10000.0*AIWR))*T+0.5                        00009990
      IPOT(83 )=(60.0*AKT4/(10000.0*AIBR))*T                            00009990
C     IPOT(177)=TOU*2.0*CRP*T+0.5                                       00010000
      IPOT(29 )=.1*TOU*2.0*CRP*T                                        00010000
C     IPOT(178)=IPOT(177)                                               00010010
      IPOT(54 )=IPOT(29 )                                               00010010
C     IPOT(179)=IPOT(135)                                               00010020
      IPOT(107)=IPOT(87 )                                               00010020
C     IPOT(180)=IPOT(136)                                               00010030
      IPOT(108)=IPOT(88 )                                               00010030
C     IPOT(181)=IPOT(135)                                               00010040
      IPOT(117)=IPOT(87 )                                               00010040
C     IPOT(182)=IPOT(136)                                               00010050
      IPOT(118)=IPOT(88 )                                               00010050
C     IPOT(183)=1.5*T+0.5                                               00010060
      IPOT(106)=0.2*T                                                   00010060
C     IPOT(184)=1.5*T+0.5                                               00010070
      IPOT(116)=IPOT(106)                                               00010070
C     IPOT(192)=(AKT3/(5000.0*AMUR))*T+0.5                              00010080
      IPOT(10 )=(AKT3/(5000.0*AMUR))*T                                  00010080
C     IPOT(193)=(AKT4/(5000.0*AMUR))*T+0.5                              00010090
      IPOT(11 )=(AKT4/(5000.0*AMUR))*T                                  00010090
C     IPOT(194)=(0.2/AMS)*T+0.5                                         00010100
      IPOT(12 )=(0.1/AMS)*T                                             00010100
C     IPOT(195)=(0.1/AMUR)*T+0.5                                        00010110
      IPOT(13 )=(0.1/AMUR)*T                                            00010110
C     IPOT(196)=IPOT(195)                                               00010120
      IPOT(09 )=IPOT(13 )                                               00010120
C     IPOT(176)=TOU*CRP*T+0.5                                           00010130
      IPOT(27 )=TOU*CRP*T                                               00010130
C     IPOT(188)=IPOT(176)                                               00010140
      IPOT(49 )=IPOT(27 )                                               00010140
      IF(IAX.EQ.2)GO TO 1021                                            00010150
C###########################################
C#####     SOLID AXLE LOGIC FOR 680      #####
C###########################################
              CALL SSRP(00,IRLERR)
              CALL SSRP(01,IRLERR)
              CALL SSRP(02,IRLERR)
              CALL SSRP(03,IRLERR)
              CALL SSRP(07,IRLERR)
              CALL SSRP(10,IRLERR)
              CALL RSCL( 01,IRSCLE )
C     IPOT(93)=0                                                        00010160
      IPOT(41)=0                                                        00010160
C     IPOT(95)=0                                                        00010170
      IPOT(40)=0                                                        00010170
C     IPOT(103)=0                                                       00010180
      IPOT(42 )=0                                                       00010180
C     IPOT(109)=0                                                       00010190
      IPOT(81 )=0                                                       00010190
C     IPOT(111)=0                                                       00010200
      IPOT(80 )=0                                                       00010200
C     IPOT(119)=0                                                       00010210
      IPOT(82 )=0                                                       00010210
C     IPOT(155)=(TS02/40.0)*T+0.5                                       00010220
      IPOT(36 )=(TS02/40.0)*T                                           00010220
C     IPOT(163)=(HUN*TS02*T+0.5)                                        00010230
      IPOT(46 )=(HUN*TS02*T)                                            00010 30
C     IPOT(167)=(TR02/40.0)*T+0.5                                       00010240
      IPOT(109)=(TR02/40.0)*T                                           00010240
C     IPOT(168)=IPOT(167)                                               00010250
      IPOT(115)=IPOT(109)                                               00010250
C     IPOT(171)=IPOT(155)                                               00010260
      IPOT(45 )=IPOT(36 )                                               00010260
C     IPOT(172)=IPOT(163)                                               00010270
      IPOT(35 )=IPOT(46 )                                               00010270
C     IPOT(173)=(RR/(4000.0*TS))*T+0.5                                  00010280
      IPOT(26 )=(RR/(4000.0*TS))*T                                      00010280
C     IPOT(185)=IPOT(173)                                               00010290
      IPOT(53 )=IPOT(26 )                                               00010290
C     IPOT(189)=0.40*T+0.5                                              00010300
      IPOT(17 )=0.40*T                                                  00010300
C     IPOT(191)=(10.0*TS/AIR)*T+0.5                                     00010310
      IPOT(15 )=(10.0*TS/AIR)*T                                         00010310
      GO TO 1022                                                        00010320
 1021 CONTINUE                                                          00010330
C###########################################
C#####     SPLIT AXLE LOGIC FOR 680      #####
C###########################################
              CALL SSRM(00,IRLERR)
              CALL SSRM(01,IRLERR)
              CALL SSRM(02,IRLERR)
              CALL SSRM(03,IRLERR)
              CALL SSRM(07,IRLERR)
              CALL SSRM(10,IRLERR)
              CALL SSCL( 01,ISSCLE )
C     IPOT(93)=2.0/(10.0*AMUR)*T+0.5                                    00010340
      IPOT(41)=2.0/(10.0*AMUR)*T                                        00010340
C     IPOT(95)=AKT3*2.0/(5000.0*AMUR)*T+0.5                             00010350
      IPOT(40)=AKT3*2.0/(5000.0*AMUR)*T                                 00010350
C     IPOT(103)=(0.2/AMS)*T+0.5                                         00010360
```

```
      IPOT(42 )=(0.1/AMS)*T                                             00010360
C     IPOT(109)=2.0/(10.0*AMUR)*T+0.5                                   00010370
      IPOT(81 )=2.0/(10.0*AMUR)*T                                       00010370
C     IPOT(111)=AKT4*2.0/(5000.0*AMUR)*T+0.5                            00010380
      IPOT(80 )=AKT4*2.0/(5000.0*AMUR)*T                                00010380
C     IPOT(119)=(0.2/AMS)*T+0.5                                         00010390
      IPOT(82 )=(0.1/AMS)*T                                             00010390
C     IPOT(155)=0                                                       00010400
      IPOT(36 )=0                                                       00010400
C     IPOT(163)=0                                                       00010410
      IPOT(46 )=0                                                       00010410
C     IPOT(167)=0                                                       00010420
      IPOT(109)=0                                                       00010420
C     IPOT(168)=0                                                       00010430
      IPOT(115)=0                                                       00010430
C     IPOT(171)=0                                                       00010440
      IPOT(45 )=0                                                       00010440
C     IPOT(172)=0                                                       00010450
      IPOT(35 )=0                                                       00010450
C     IPOT(173)=0                                                       00010460
      IPOT(26 )=0                                                       00010460
C     IPOT(185)=0                                                       00010470
      IPOT(53 )=0                                                       00010470
C     IPOT(189)=0                                                       00010480
      IPOT(17 )=0                                                       00010480
C     IPOT(191)=0                                                       00010490
      IPOT(15 )=0                                                       00010490
 1022 CONTINUE                                                          00010500
C     SFIN(1)=-100.0*TIN                                                00010510
      SFIN(1)=-100.0                                                    00010510
      SFIN(2)=SFIN(1)                                                   00010520
      SFIN(3)=SFIN(1)                                                   00010530
C     SFIN(4)=10.0*TIN                                                  00010540
      SFIN(4)=10.0                                                      00010540
      SFIN(5)=SFIN(4)                                                   00010550
      SFIN(6)=SFIN(4)                                                   00010560
C     SFIN(7)=TIN                                                       00010570
      SFIN(7)=1.0                                                       00010570
C     SFIN(8)=0.25*TIN                                                  00010580
      SFIN(8)=0.25                                                      00010580
      IF(IAX.EQ.2)SFIN(7)=SFIN(1)                                       00010590
      IF(IAX.EQ.2)SFIN(8)=SFIN(4)                                       00010600
C     SFIN(9)=0.5*TIN                                                   00010610
      SFIN(9)=0.5                                                       00010610
      SFIN(10)=SFIN(9)                                                  00010620
C     SFIN(11)=2.0/10000.0                                              00010630
      SFIN(11)=2.0                                                      00010630
      SFIN(12)=SFIN(11)                                                 00010640
      SFIN(13)=SFIN(11)                                                 00010650
      SFIN(14)=SFIN(11)                                                 00010660
C     SFIN(15)=1000.0*TIN                                               00010670
      SFIN(15)=1000.0                                                   00010670
      SFIN(16)=SFIN(15)                                                 00010680
      SFIN(17)=SFIN(15)                                                 00010690
      SFIN(18)=SFIN(15)                                                 00010700
C     SFIN(19)=0.0001                                                   00010710
      SFIN(19)=1.0                                                      00010710
C     SFIN(20)=0.0001                                                   00010720
      SFIN(20)=1.0                                                      00010720
C     SFIN(21)=0.0001                                                   00010730
```

```
      SFIN(21)=1.0                                                      00010730
C     SFIN(22)=0.0001                                                   00010740
      SFIN(22)=1.0                                                      00010740
C     SFIN(23)=100./T                                                   00010750
      SFIN(23)=100.                                                     00010750
C     SFIN(24)=100./T                                                   00010760
      SFIN(24)=100.                                                     00010760
C     SFIN(25)=1.0/T                                                    00010770
      SFIN(25)=1.0                                                      00010770
C     SFIN(26)=1.0/T                                                    00010780
      SFIN(26)=1.0                                                      00010780
C     SFIN(27)=1.0/T                                                    00010790
      SFIN(27)=1.0                                                      00010790
C     SFIN(28)=1.0/T                                                    00010800
      SFIN(28)=1.0                                                      00010800
      IN(23)=RPS1+SIGN(0.5,RPS1)                                        00010810
      IN(24)=RPS2+SIGN(0.5,RPS2)                                        00010820
      IN(25)=0                                                          00010830
C     IN(26)=10000                                                      00010840
      IN(26)=1                                                          00010840
      IN(27)=0                                                          00010850
C     IN(28)=10000                                                      00010860
      IN(28)=1                                                          00010860
      DO 10 I=1,48                                                      00010870
      SFOUT(I)=1.0                                                      00010880
   10 CONTINUE                                                          00010890
C     SFOUT(9)=1.0                                                      00010900
      SFOUT(9)=1.0/10000.                                               00010900
C     SFOUT(10)=0.1*T                                                   00010910
      SFOUT(10)=0.1                                                     00010910
C     SFOUT(14)=1.0                                                     00010920
      SFOUT(14)=1.0/10000.                                              00010920
C     SFOUT(25)=0.02*T*AKT3/AIR                                         00010930
      SFOUT(25)=0.02*  AKT3/AIR                                         00010930
C     SFOUT(27)=0.02*T*AKT4/AIR                                         00010940
      SFOUT(27)=0.02*  AKT4/AIR                                         00010940
C     SFOUT(28)=0.01*T                                                  00010950
      SFOUT(28)=0.01                                                    00010950
C     SFOUT(37)=(1.0/1500.0)*T                                          00010960
      SFOUT(37)=(1.0/1500.0)                                            00010960
      RETURN                                                            00010970
      END                                                               00010980
```

```
C     SUBROUTINE SBPG2                                                  00012209
      SUBROUTINE SBPG2                                                  00012210
      COMMON/APL/ OPEN  ,RTSW  ,LDTSW ,RBSW
      COMMON/DEVICE/KEYBD,ITTY,ICDRD,LPTR
      COMMON/HHHH/H1,H2,H3,H4                                           00012220
      COMMON/SPLTAX/SPSR3,CPSR3,SPSR4,CPSR4,SCR3,SCR4,TBCR3,TBCR4,TBSR3,00012230
     1   TBSR4,TRSR3,TRSR4,TRCR3,TRCR4,PSR3,PSR4,IAX                    00012240
      COMMON/TABLES/PHIFNT(7),THEFNT(7),PSIFNT(7),PHIRR(7),THERR(7),    00012250
     1   PSIRR(7)                                                       00012260
      COMMON/EXTRA/             UOUT,ROUT,        PSIOUT,EXTAB(100),    00012270
     1   DLYTB(100),RTAB(100)                                           00012280
      COMMON/FRIDAY/BTVMAX,BTVTB(100)                                   00012290
      COMMON/THINGS/TMAX1,TMAX2,TMAX3,TQRMAX,TQFMAX,PSIMAX,ONER
      COMMON/FEES/FEE1,FEE2,THE1,THE2                                   00012310
      COMMON/DELS/DELSWC                                                00012320
      COMMON/CLEAN/ONEOA,ONEOD,PBSW,PB                                  00012330
      COMMON/NUTS/TOO                                                   00012340
      COMMON/XYZ/NUMBR                                                  00012350
      COMMON/QUANTS/QUAN1,QUAN2,QUAN3,QUAN4                             00012360
      COMMON/EFFS/ANUM,ADEN,ANUMDT,ADENDT,ANUMO,ADENO,ANUMDO,ADENDO,    00012370
     1   ANOUT,ADOUT                                                    00012380
      COMMON/XBS/XB(12),NS(4,12),DELX(4),XI(4),NNN                      00012390
      COMMON/NEWER/TIME25,TIME10,RMAX,ETAMAX,PSI5,PHIMAX,DSWMAX         00012400
      COMMON/VARS/P,Q,R,U,V,W,X,Y,Z,THE,PHI,PSI,PO,QO,RO,UO,VO,WO,XO,   00012410
     1     YO,ZO,THEO,PHIO,PSIO                                         00012420
      COMMON/UVW/VC,UIN                                                 00012430
      COMMON/EES/O1,O2,O3,E4,E5,E6                                      00012440
      COMMON/ZILCH/TQMAXP,AKTQP                                         00012450
      COMMON/INOUT/INA(32),IOUTA(48),IN(32),IOUT(48),ISW1,ISW7,SFIN(32),00012460
     1      SFOUT(48),IPRT,ITMP(48)                                     00012470
      COMMON/ANALOG/DEL1DT,DEL2DT,DEL3DT,DEL1,DEL2,DEL3,PHIRD,PHIR,     00012480
     1     DELFW1,DELFW2,U1P,U2P,U3P,U4P,S1P,S2P,S3P,S4P,TTA,PFSLP      00012490
      COMMON/COMBLK/AIXP,SM,AIYP,AIXZP,GAM1,GAM2,GAM3,AIXBR,AIYBR,      00012500
     1     AIZBR,A1,A2,AIXZBR,A12,E1,E2,E3,DELTA,GV1,GV2,GP1,GP2,GR1,   00012510
     1     GR2,CIP,CIVP,RZF,RZR,A2T,CA20,CA23,TIME,DT,      ANGNL,ANGNLO 00012520
     1     ,TR02,TF02,TS02,G,THRD,TWN7                                  00012530
      COMMON/DDD/DIN,DDTO,DDT,TIME5,DDO,DD,DOUT                         00012540
      COMMON/OPSW/IHSW                                                  00012550
      COMMON/SP7BLK/N1,N2,IPOT(244),IPOTAD(244),PARAM(300)              00012560
      COMMON/MICKEY/IRUNTB(100),VTB(100),SNLTB(100),SNRTB(100),DSWTB(10000012570
     1  ),BTTB(100),DTB(100),EFFTB(100),DYTB(100),PSITB (100),NRUN,     00012580
     1   YSPEC,PSIM,XPF                                                 00012590
     1   ,TMPTB(100)                                                    00012600
      COMMON/NONAME/XEND,O,EXIT2
      DIMENSION CSI(4),XBM(4),SLP(4)                                    00012620
      EQUIVALENCE                                                       00012630
     1   (PARAM( 1),AMS)     , (PARAM( 2),AMUF)  , (PARAM( 3),AMUR)   , 00012640
     1   (PARAM( 4),ZF)      , (PARAM( 5),ZR)    , (PARAM( 6),A)      , 00012650
     1   (PARAM( 7),B)       , (PARAM( 8),TF)    , (PARAM( 9),TR)     , 00012660
     1   (PARAM(10),TS)      , (PARAM(11),AIX)   , (PARAM(12),AIY)    , 00012670
     1   (PARAM(13),AIZ)     , (PARAM(14),AIXZ)  , (PARAM(15),AIR)    , 00012680
     1   (PARAM(16),CF)      , (PARAM(17),RF)    , (PARAM(18),CFP)    , 00012690
     1   (PARAM(19),AKF)     , (PARAM(20),ALAMF) , (PARAM(21),OFC)    , 00012700
     1   (PARAM(25),CRP)     , (PARAM(26),AKR)   , (PARAM(27),ALAMR)  , 00012710
     1   (PARAM(22),OFT)     , (PARAM(23),CR)    , (PARAM(24),RR)     , 00012720
     1   (PARAM(28),ORC)     , (PARAM(29),ORT)   , (PARAM(30),AKRS)   , 00012730
     1   (PARAM(31),RW)      , (PARAM(32),B4)    , (PARAM(33),OT)     , 00012740
     1   (PARAM(34),CA0)     , (PARAM(35),CA1)   , (PARAM(36),CA2)    , 00012750
     1   (PARAM(37),CA3)     , (PARAM(38),CA4)   , (PARAM(39),AISW)   , 00012760
```

```
1    (PARAM(44),AKDL)    , (PARAM(41),AKSC)   , (PARAM(42),ANG)     ,   00012770
1    (PARAM(43),WG)      , (PARAM(40),ANL2)   , (PARAM(45),AKSL)        00012780
 EQUIVALENCE                                                            00012790
1    (PARAM(46),ANL1)    , (PARAM(47),AIFW)   , (PARAM(48),HDL)     ,   00012800
1    (PARAM(49),AIWF)    , (PARAM(50),AIWR)   , (PARAM(51),AID)     ,   00012810
1    (PARAM(52),ARBR)    , (PARAM(53),EPS1)   , (PARAM(54),EPS2)    ,   00012820
1    (PARAM(55),PTBR)    , (PARAM(56),YSA1)   , (PARAM(57),YSA2)    ,   00012830
1    (PARAM(58),YHS1)    , (PARAM(59),YHS2)   , (PARAM(60),AKD)     ,   00012840
1    (PARAM(61),TODBR)   , (PARAM(62),AK)     , (PARAM(63),PIN)     ,   00012850
1    (PARAM(64),QIN)     , (PARAM(65),RIN)    , (PARAM(66),UIZ)     ,   00012860
1    (PARAM(67),VIN)     , (PARAM(68),WIN)    , (PARAM(69),XIN)     ,   00012870
1    (PARAM(70),YIN)     , (PARAM(71),ZIN)    , (PARAM(72),THEIN)   ,   00012880
1    (PARAM(73),PHIIN)   , (PARAM(74),PSIIN)  , (PARAM(75),DTIN)    ,   00012890
1    (PARAM(76),TEND)    , (PARAM(77),AKT1)   , (PARAM(78),AKT2)  ,     00012900
1    (PARAM(79),AKT3)    , (PARAM(80),AKT4)   , (PARAM(81),RPS1)  ,     00012910
1    (PARAM(82),RPS2)    , (PARAM(83),RPS3)   , (PARAM(84),RPS4)    ,   00012920
1    (PARAM(85),B1)      , (PARAM(86),B2)     , (PARAM(87),B3)          00012930
 EQUIVALENCE                                                            00012940
1                          (PARAM( 89),DEL1DN),(PARAM( 90),DEL2DN),     00012950
1    (PARAM( 91),DEL3DN),(PARAM( 92),DEL1IN), (PARAM( 93),DEL2IN),      00012960
1    (PARAM( 94),DEL3IN), (PARAM( 95),PHIDN), (PARAM(96),PHIRN),        00012970
1    (PARAM( 97),DFW1IN), (PARAM( 98),DFW2IN),(PARAM( 99),U1PIN),       00012980
1    (PARAM(100),U2PIN),  (PARAM(101),U3PIN), (PARAM(102),U4PIN),       00012990
1    (PARAM(103),S1PIN),  (PARAM(104),S2PIN), (PARAM(105),S3PIN),       00013000
1    (PARAM(106),S4PIN), (PARAM(107),PPRT)                              00013010
1,(PARAM(110),TQMAX),(PARAM(111),AKTQ),(PARAM(112),VCIN)                00013020
1,(PARAM(113),SWMT),(PARAM(114),DSWCM),(PARAM(115),TST),                00013030
1(PARAM(116),DSLP),(PARAM(117),CGAM),(PARAM(118),CS)                    00013040
1   ,(PARAM(119),TQRBR),(PARAM(120),TQFBR)                              00013050
1   ,(PARAM(121),PFL),(PARAM(122),TTD),(PARAM(123),DSW)                 00013060
1     ,(PARAM(124),TSW)                                                 00013070
 EQUIVALENCE                                                            00013080
1   (PARAM(130),AMCR),(PARAM(131),ESP),(PARAM(132),AKSL1),              00013090
1   (PARAM(133),AKSL2),(PARAM(134),AA1),(PARAM(135),AA2),               00013100
1   (PARAM(136),CCR),(PARAM(137),CFCR),(PARAM(138),AP),                 00013110
1   (PARAM(139),EP1),(PARAM(140),EP2),(PARAM(141),ERR1),                00013120
1   (PARAM(142),ERR2),                                                  00013130
1   (PARAM(143),AML1),(PARAM(144),AML2),(PARAM(145),RRIM),              00013140
1    (PARAM(146),RWR),(PARAM(147),AKV),                                 00013150
1     (PARAM(148),AK1 ) , (PARAM(149),AK2 ) , (PARAM(150),AM   ) ,      00013160
1     (PARAM(151),PM   ) , (PARAM(152),PV   ) , (PARAM(153),RT   ) ,    00013170
1     (PARAM(154),TD1 ) , (PARAM(155),TD2 ) , (PARAM(156),TD3 ) ,       00013180
1     (PARAM(157),TD4 ) , (PARAM(158),Y1   ) , (PARAM(159),YMAX) ,      00013190
1     (PARAM(160),ZET1) , (PARAM(161),ZET2) , (PARAM(162),EX1 ) ,       00013200
1     (PARAM(163),TBF ) , (PARAM(164),TBR ) , (PARAM(165),T1   ) ,      00013210
1     (PARAM(166),WN1 ) , (PARAM(167),WN2 ) , (PARAM(168),R1   ) ,      00013220
1     (PARAM(169),PA0 )                                                 00013230
1    ,(PARAM(196),EPSK1),(PARAM(197),EPSK2)                             00013240
1   ,(PARAM(274),TLF),(PARAM(275),ZR0),(PARAM(276),ALF)                 00013250
 EQUIVALENCE                                                            00013260
1    (PARAM(223),CR1C),(PARAM(224),CR1T),(PARAM(225),CR2C),             00013270
1    (PARAM(226),CR2T),(PARAM(227),CR3C),(PARAM(228),CR3T),             00013280
1    (PARAM(229),CR4C),(PARAM(230),CR4T),(PARAM(231),AH1),              00013290
1    (PARAM(232),AH2),(PARAM(233),ANL),(PARAM(234),AKF1),               00013300
1   (PARAM(235),AKF2),(PARAM(236),AKR3),(PARAM(237),AKR4)               00013310
 EQUIVALENCE                                                            00013320
1    (PARAM(281),TSF),(PARAM(282),TSR),(PARAM(283),TANP),               00013330
1    (PARAM(284),HFC),(PARAM(285),HRC),(PARAM(286),CRRC)                00013340
 EQUIVALENCE                                                            00013350
1    (PARAM(290),ROT),(PARAM(291),RA0),(PARAM(292),RA1),
     1    (PARAM(293),RA2),(PARAM(294),RA3),(PARAM(295),RA4)
      EQUIVALENCE  (PHIRD,DEL4DT),(PHIR,DEL4)                           00013350
      DATA T/10000.0/                                                   00013360
      REAL*4       IOUT  ,IN    ,ITMP
      INTEGER*2    RTSW  ,RBSW  ,LDTSW ,OPEN  ,OPNN
C        LI,7      X'4000'                                              00013370
C     WD,7 X'1202'                                                      00013380
C     CALL  STBLK(IOUTA,IOUT,48)                                        00013390
C     CALL  SCANH(INA,IN,28)                                            00013400
C     CALL  LBDA(0,48,IOUT,ILBDAE)                                      00013390
C     CALL  TLDA                                                        00013391
C     CALL  CRBC(0,28,IN,ICRBDE)                                        00013400
      DEL1DT= IN( 1)*SFIN( 1)                                           00013410
      DEL2DT= IN( 2)*SFIN( 2)                                           00013420
      DEL3DT= IN( 3)*SFIN( 3)                                           00013430
      DEL1  = IN( 4)*SFIN( 4)                                           00013440
      DEL2  = IN( 5)*SFIN( 5)                                           00013450
      DEL3=   IN( 6)*SFIN( 6)                                           00013460
      PHIRD=  IN( 7)*SFIN( 7)                                           00013470
      PHIR=   IN( 8)*SFIN( 8)                                           00013480
C     DELFW1= IN( 9)*SFIN( 9)                                           00013490
      DELFW1=-IN( 9)*SFIN( 9)                                           00013490
C     DELFW2= IN(10)*SFIN(10)                                           00013500
      DELFW2=-IN(10)*SFIN(10)                                           00013500
      U1P=    IN(11)*SFIN(11)                                           00013510
      U2P=    IN(12)*SFIN(12)                                           00013520
      U3P=    IN(13)*SFIN(13)                                           00013530
      U4P=    IN(14)*SFIN(14)                                           00013540
      S1P=    IN(15)*SFIN(15)                                           00013550
      S2P=    IN(16)*SFIN(16)                                           00013560
      S3P=    IN(17)*SFIN(17)                                           00013570
      S4P=    IN(18)*SFIN(18)                                           00013580
      QUAN1=IN(19)*SFIN(19)+1.0                                         00013590
      QUAN2=IN(20)*SFIN(20)+1.0                                         00013600
      QUAN3= IN(21)*SFIN(21)+1.0                                         0001361
      QUAN4= IN(22)*SFIN(22)+1.0                                         0001362
      ARPS1= IN(23)*SFIN(23)                                             0001363
      ARPS2= IN(24)*SFIN(24)                                             000 364
      WSTH1= IN(25)*SFIN(25)                                             0001365
      WCTH1= IN(26)*SFIN(26)                                             0001366
      WSTH2= IN(27)*SFIN(27)                                             0001367
      WCTH2= IN(28)*SFIN(28)                                             0001368
      IHSW=1                                                            00013690
         FXIT2  = 10.0*5280.0*12.0/3600.0                               00013691
      ENTRY SBPG22                                                      00013700
      IF(TIME.EQ.0.)TIMEB=0.                                            00013710
      M=1                                                               00013720
      IF(TIME.GE.PARAM(175))M=0                                         00013730
C     CALL  SETWD(0,M)                                                  00013740
C 998 FORMAT('0'8E15.6/(' '8E15.6))                                     00013750
  998 FORMAT('0',8E15.6/(' ',8E15.6))                                      13750
      ISW=1                                                             00013760
      IF(TIME.EQ.0.0)FS1=0.0                                            00013770
      IF(TIME.EQ.0.0)FS2=0.0                                            00013780
      IF(IPRT.EQ.0)GO TO 100                                            00013790
      WRITE(ITTY ,998)DEL1DT,DEL2DT,DEL3DT,DEL1,DEL2,DEL3,PHIRD,PHIR,     000138
     1    DELFW1,DELFW2,U1P,U2P,U3P,U4P,S1P,S2P,S3P,S4P                 00013810
     1   ,QUAN1,QUAN2,QUAN3,QUAN4,PB                                    00013820
  100 TM1=Z-A*THE                                                       00013830
  999 FORMAT('0AND THERE YOU ARE IN SBPG2')                             00013840
```

```
      IF(PB.GT.50.)PBSW=1.
      TM2=TF02*PHI
      Z1PP=TM1+TM2
      Z1P=Z1PP+ZF
      Z1=Z1P+DEL1
      Z2PP=TM1-TM2
      Z2P=Z2PP+ZF
      Z2=Z2P+DEL2
      TM1=Z+B*THE
      TM2=TR02*PHI
      Z3PP=TM1+TM2
      Z3P=Z3PP+ZR
      Z3=Z3P+DEL3
      IF(IAX.EQ.1)Z3=Z3+TR02*PHIR+DEL3-DEL3
      Z4PP=TM1-TM2
      Z4P=Z4PP+ZR
      Z4=Z4P+DEL4
      IF(IAX.EQ.1)Z4=Z4-TR02*PHIR+DEL3-DEL4
      RWZ1=RW+Z1
      RWZ2=RW+Z2
      RWZ3=RW+Z3
      RWZ4=RW+Z4
      FR1=0.0
      FR2=0.0
      FR3=0.0
      FR4=0.0
      IF(RWZ1.GT.0.0)FR1=AKT1*RWZ1
      IF(RWZ2.GT.0.0)FR2=AKT2*RWZ2
      IF(RWZ3.GT.0.0)FR3=AKT3*RWZ3
      IF(RWZ4.GT.0.0)FR4=AKT4*RWZ4
      TM1=U+ZF*Q
      TM2=TF02*R
      U1=TM1-TM2
      U2=TM1+TM2
      TM1=U+ZR*Q
      TM2=TR02*R
      U3=TM1-TM2
      U4=TM1+TM2
      VARZFP=V+A*R-ZF*P
      VBRZRP=V-B*R-ZR*P
      PPHIR=P
      IF(IAX.EQ.1)PPHIR=P+PHIRD
      V1=VARZFP+Z1*P
      V2=VARZFP+Z2*P
      V3=VBRZRP+Z3*PPHIR
      V4=VBRZRP+Z4*PPHIR
      WAQ=W-A*Q
      WBQD3=W+B*Q+DEL3DT
      TF2P=TF02*P
      TR2P=TR02*P
      W1=WAQ+TF2P+DEL1DT
      W2=WAQ-TF2P+DEL2DT
      IF(IAX.EQ.2)GO TO 9738
      W3=WBQD3+TR02*PPHIR
      W4=WBQD3-TR02*PPHIR
      GO TO 9739
 9738 WBQD4=W+B*Q+DEL4DT
      W3=WBQD3+TR02*P
      W4=WBQD4-TR02*P
 9739 CONTINUE
```

```
      PSI1=DELEW1+POLY(DEL1,PSIFNT)+EPSK1                               00014450
      PSI2=DELEW2-POLY(DEL2,PSIFNT)+EPSK2                               00014460
      PSI3=AKRS*PHIR                                                    00014470
      PSI4=PSI3                                                         00014480
      PHI1=POLY(DEL1,PHIFNT)+SIGN(FEE1,FS1)                             00014490
      PHI2=-POLY(DEL2,PHIFNT)+SIGN(FEE2,FS2)                            00014500
      PHI3=PHIR                                                         00014510
      PHI4=PHIR                                                         00014520
      IF(IAX.EQ.1)GO TO 7843                                            00014530
      PSI3=POLY(DEL3,PSIRR)                                             00014540
      PSI4=-POLY(DEL4,PSIRR)                                            00014550
      PHI3 =POLY(DEL3,PHIRR)                                            00014560
      PHI4=-POLY(DEL4,PHIRR)                                            00014570
 7843 UG1=U1+THE*W1                                                     00014590
      UG2=U2+THE*W2                                                     00014600
      UG3=U3+THE*W3                                                     00014610
      UG4=U4+THE*W4                                                     00014620
      VG1=V1-PHI*W1                                                     00014630
      VG2=V2-PHI*W2                                                     00014640
      VG3=V3-PHI*W3                                                     00014650
      VG4=V4-PHI*W4                                                     00014660
C     SINPS1=XYSIN(PSI1)                                                00014670
      SINPS1=SIN(PSI1)                                                  00014670
C     SINPS2=XYSIN(PSI2)                                                00014680
      SINPS2=SIN(PSI2)                                                  00014680
      SINPS3=PSI3                                                       00014690
C     SINPS4=SINPS3                                                     00014700
      SINPS4=PSI4                                                       00014700
C     COSPS1=XYCOS(PSI1)                                                00014710
      COSPS1=COS(PSI1)                                                  00014710
C     COSPS2=XYCOS(PSI2)                                                00014720
      COSPS2=COS(PSI2)                                                  00014720
      COSPS3=1.0                                                        00014730
      COSPS4=COSPS3                                                     00014740
      UG1P=UG1*COSPS1+VG1*SINPS1                                        00014750
      UG2P=UG2*COSPS2+VG2*SINPS2                                        00014760
      UG3P=UG3*COSPS3+VG3*SINPS3                                        00014770
      UG4P=UG4*COSPS4+VG4*SINPS4                                        00014780
      PHICG1=THE*SINPS1+PHI*COSPS1+PHI1                                 00014790
      PHICG2=THE*SINPS2+PHI*COSPS2+PHI2                                 00014800
      PHICG3=0.0                                                        00014810
      PHICG4=0.0                                                        00014820
      IF(IAX.EQ.1) GO TO 900
      PHICG3=THE*SINPS3+PHI*COSPS3+PHI3
      PHICG4=THE*SINPS4+PHI*COSPS4+PHI4
  900 CV1=SQRT(U*U+V*V)*3600.0/(12.0*5280.0)                            00014830
      CV2=CV1                                                           00014840
      CV3=CV1                                                           00014850
      CV4=CV1                                                           00014860
      AB1=ABS(UG1)                                                      00014870
      AB2=ABS(UG2)                                                      00014880
      AB3=ABS(UG3)                                                      00014890
      AB4=ABS(UG4)                                                      00014900
      BETA1=ATAN(VG1/AB1)-PSI1*UG1/AB1                                  00014910
      BETA2=ATAN(VG2/AB2)-PSI2*UG2/AB2                                  00014920
      BETA3=ATAN(VG3/AB3)-PSI3*UG3/AB3                                  00014930
      BETA4=ATAN(VG4/AB4)-PSI4*UG4/AB4                                  00014940
      AMU1=FNCT(B1,B2,B3,FR1,CV1)                                       00014950
      AMU2=FNCT(B1,B2,B3,FR2,CV2)                                       00014960
      AMU3=FNCT(B1,B2,B3,FR3,CV3)                                       00014970
      AMU4=FNCT(B1,B2,B3,FR4,CV4)
      CZ=COS(PSI)
      SN=SIN(PSI)
      SN1=PARAM(170)
      SN2=PARAM(170)
      SN3=PARAM(170)
      SN4=PARAM(170)
      INTFUN=PARAM(172)+0.5
      IF(INTFUN.EQ.0)GO TO 3497
      IF(INTFUN.NE.1)GO TO 3498
      X1=A*CZ-TE02*SN   +X
      X2=A*CZ+TE02*SN   +X
      X3=-B*CZ-TR02*SN +X
      X4=-B*CZ+TR02*SN+X
      SN1=PARAM(170)
      SN2=PARAM(170)
      SN3=PARAM(170)
      SN4=PARAM(170)
      TEMP=PARAM(173)+PARAM(174)
      TEMP=TEMP*12.0
      PPPP=PARAM(173)*12.0
      IF(X1.GT.PPPP       .AND.X1.LE.TEMP)SN1=PARAM(171)
      IF(X2.GT.PPPP       .AND.X2.LE.TEMP)SN2=PARAM(171)
      IF(X3.GT.PPPP       .AND.X3.LE.TEMP)SN3=PARAM(171)
      IF(X4.GT.PPPP       .AND.X4.LE.TEMP)SN4=PARAM(171)
      GO TO 3498
 3497 CONTINUE
      YY1=A*SN+TE02*CZ   +Y
      Y2=A*SN-TE02*CZ    +Y
      Y3=-B*SN+TR02*CZ +Y
      Y4=-B*SN-TR02*CZ+Y
      SN1=PARAM(170)
      SN2=PARAM(170)
      SN3=PARAM(170)
      SN4=PARAM(170)
      IF(YY1.LT.0.0)SN1=PARAM(171)
      IF(Y2.LT.0.0)SN2=PARAM(171)
      IF(Y3.LT.0.0)SN3=PARAM(171)
      IF(Y4.LT.0.0)SN4=PARAM(171)
 3498 CONTINUE
      AMU1=AMU1*SN1*0.025
      AMU2=AMU2*SN2*0.025
      AMU3=AMU3*SN3*0.025
      AMU4=AMU4*SN4*0.025
  600 CONTINUE
      RM1=FR1*AMU1
      FR1BR=AMIN(FR1,A2T)
      ALF1=CA1*FR1BR*(FR1BR-CA2)-CA20
      IF( ALF1/CA2.GE.0.0 ) ALF1=1.0E-10
      BET1P =CA23*(CA4-FR1BR)*FR1BR*PHICG1/(CA4*ALF1)
      IF(RM1.EQ.0.0)GO TO 610
      BET1BR=ALF1*(BETA1+BET1P)/(CA2*RM1)
      GO TO 710
  610 RM1P=0.0
      BET1BR=0.0
  710 CONTINUE
      RM2=FR2*AMU2
      FR2BR=AMIN(FR2,A2T)
      ALF2=CA1*FR2BR*(FR2BR-CA2)-CA20
      IF( ALF2/CA2.GE.0.0 ) ALF2=1.0E-10
      BET2P =CA23*(CA4-FR2BR)*FR2BR*PHICG2/(CA4*ALF2)                   00015560
      IF(RM2.EQ.0.0)GO TO 620                                           00015570
      BET2BR=ALF2*(BETA2+BET2P)/(CA2*RM2)                               00015580
      GO TO 720                                                         00015590
  620 RM2P=0.0                                                          00015600
      BET2BR=0.0                                                        00015610
  720 CONTINUE                                                          00015620
      RA23=RA2*RA3
      R2T=ROT*RA2
      RA20=RA0*RA2
      RM3=FR3*AMU3                                                      00015630
C     FR3BR=AMIN(FR3,A2T)                                               00015640
      FR3BR=AMIN( FR3,R2T )
C     ALF3=CA1*FR3BR*(FR3BR-CA2)-CA20                                   00015650
      ALF3=RA1*FR3BR*(FR3BR-RA2)-RA20
      IF( ALF3/RA2.GE.0.0 ) ALF3=1.0E-10
      BET3P=RA23*(RA4-FR3BR)*FR3BR*PHICG3/(RA4*ALF3)
      IF( RM3.EQ.0.0 ) GO TO 630
      BET3BR=ALF3*(BETA3+BET3P )/(RA2*RM3)
C     IF( IAX.EQ.1 ) GO TO 15
C        BET3P  = CA23*( CA4-FR3BR )*FR3BR*PHICG3 / ( CA4*ALF3 )
C     GO TO 16
C  15 BET3P = 0.0
C  16 CONTINUE
C     IF(RM3.EQ.0.0)GO TO 630                                           00015670
C     BET3BR=ALF3*(BETA3+BET3P)/(CA2*RM3)                               00015680
      GO TO 730                                                         00015690
  630 RM3P=0.0                                                          00015700
      BET3BR=0.0                                                        00015710
  730 CONTINUE                                                          00015720
C 996 FORMAT('0      'E15.6)                                            00015730
  996 FORMAT('0       ',E15.6)                                             15730
      RM4=FR4*AMU4                                                      00015740
C     FR4BR=AMIN(FR4,A2T)                                               00015750
      FR4BR=AMIN(FR4,R2T)
C     ALF4=CA1*FR4BR*(FR4BR-CA2)-CA20                                   00015760
      ALF4=RA1*FR4BR*(FR4BR-RA2)-RA20
      IF( ALF4/RA2.GE.0.0 ) ALF4=1.0E-10
      BET4P=RA23*(RA4-FR4BR)*FR4BR*PHICG4/(RA4*ALF4)
C     IF( IAX.EQ.1 ) GO TO 17
C        BET4P  = CA23*( CA4-FR4BR )*FR4BR*PHICG4 / ( CA4*ALF4 )
C     GO TO 18
C  17 BET4P  = 0.0
C  18 CONTINUE
      IF(RM4.EQ.0.0)GO TO 640                                           00015780
C     BET4BR=ALF4*(BETA4+BET4P)/(CA2*RM4)                               00015790
      BET4BR=ALF4*(BETA4+BET4P)/(RA2*RM4)
      GO TO 740                                                         00015800
  640 RM4P=0.0                                                          00015810
      BET4BR=0.0                                                        00015820
  740 CONTINUE                                                          00015830
      ALF1=ALF1*TSF                                                     00015840
      ALF2=ALF2*TSF                                                     00015850
      ALF3=ALF3*TSR                                                     00015860
      ALF4=ALF4*TSR                                                     00015870
      AB1=ABS(BET1BR)                                                   00015880
      AB2=ABS(BET2BR)                                                   00015890
      AB3=ABS(BET3BR)                                                   00015900
      AB4=ABS(BET4BR)                                                   00015910
      IF(AB1.GE.3.0)GO TO 10                                            00015920
      GB1=BET1BR*(1.0-THRD*AB1+TWN7*BET1BR**2)
   20 IF(AB2.GE.3.0)GO TO 30
      GB2=BET2BR*(1.0-THRD*AB2+TWN7*BET2BR**2)
   40 IF(AB3.GE.3.0) GO TO 50
      GB3=BET3BR*(1.0-THRD*AB3+TWN7*BET3BR**2)
   60 IF(AB4.GE.3.0)GO TO 70
      GB4=BET4BR*(1.0-THRD*AB4+TWN7*BET4BR**2)
      GO TO 80
   10 GB1=BET1BR/AB1
      GO TO 20
   30 GB2=BET2BR/AB2
      GO TO 40
   50 GB3=BET3BR/AB3
      GO TO 60
   70 GB4=BET4BR/AB4
   80 CONTINUE
      SLIP1=QUAN1
      SLIP2=QUAN2
      SLIP3=QUAN3
      SLIP4=QUAN4
      IF(QUAN1.LT.(-1.0).OR.QUAN1.GT.1.0)SLIP1=SIGN(1.0,QUAN1)
      IF(QUAN2.LT.(-1.0).OR.QUAN2.GT.1.0)SLIP2=SIGN(1.0,QUAN2)
      IF(QUAN3.LT.(-1.0).OR.QUAN3.GT.1.0)SLIP3=SIGN(1.0,QUAN3)
      IF(QUAN4.LT.(-1.0).OR.QUAN4.GT.1.0)SLIP4=SIGN(1.0,QUAN4)
      DSLM=PARAM(114)/PARAM(116)
      XTMP=PARAM(121)/PARAM(192)
      IF(PARAM(126).NE.0.0)GO TO 4321
      IF(TIME.GT.TST)GO TO 6000
      DELSWC=0.0
      GO TO 7000
 6000 DELSWC=(TIME-TST)*DSLM
      IF(ABS(DELSWC).GT.DSWCM)DELSWC=DSWCM*SIGN(1.0,DELSWC)
      IF(PARAM(128).EQ.3.0)GO TO 7000
      IF( TIME.GT.4.5 )DELSWC=DSWCM*(5.5-TIME)*SIGN(1.0,DELSWC)
 7000 DELSWC=DELSWC*.01745329
      PF=0.0
      IF(TIME.LT.TTD)GO TO 4444
      PF=(TIME-CGAM)*XTMP
      IF(PARAM(128).EQ.1.0)GO TO 2223
      IF(PARAM(128).EQ.3.0)GO TO 2223
 2222 IF(PF.GT.PFL)PF=PFL
      IF(TIME.LT.CGAM)PF=0.0
      PFR=(TIME-CS)*XTMP
      IF( TIME.GT.CS )PF=PF*(CS-TIME)/10.
      IF(TIME.LT.CS)PFR=0.0
      IF(PFR.GT.PFL)PFR=PFL
      IF( TIME.GT.CS )PFR=PFR*(CS-TIME)/10.
      TQFBR=-FF(PF)
      TQRBR=-FR(PFR)
      GO TO 2345
 2223 PF=(TIME-CGAM)*XTMP
      IF(PF.GT.PFL)PF=PFL
      PFR=(TIME-CGAM)*XTMP
      IF(PFR.GT.PFL)PFR=PFL
      TQFBR=-FF(PF)
      TQRBR=-FF(PFR)
      IF(TIME.LE.CGAM)TQFBR=0.
      IF(TIME.LE.CGAM)TQRBR=0.
      GO TO 2345
 4444 TQFBR=0.0
```

```
      TQRBR=AKTQP*(VC-UOUT)
      IF(TQRBR.GT.TQMAXP)TQRBR=TQMAXP
      GO TO 2345
 4321 CONTINUE
      IF(PARAM(178).EQ.0.)GO TO 103
      IF((TIMEB.EQ.0.).AND.(PBSW.NE.0.))TIMEB=TIME
      IF(TIMEB.EQ.0.)GO TO 104
      DELSWC=SIN(3.141593*(TIME-TIMEB)*PARAM(179))*DSW*.01745329
      IF((TIME-TIMEB).GT.TSW)DELSWC=0.
      GO TO 105
  103 CONTINUE
      DELSWC=SIN(3.141593*TIME*PARAM(179))*DSW*.01745329
      IF(TIME.GT.TSW)DELSWC=0.0
      GO TO 105
  104 DELSWC=0.
  105 CONTINUE
      PF=0.0
      TQRBR=0.0
      TQFBR=0.0
      IF(PARAM(125).EQ.0.0)GO TO 2345
      IF(TIME.LE.PARAM(278).OR.TIME.GT.PARAM(279))GO TO 2345
      PF=(TIME-PARAM(278))*26000.0
      IF(PF.GT.PFL)PF=PFL
      TQRBR=-FR(PF)
      TQFBR=-FF(PF)
 2345 CONTINUE
      SAM1=(BETA1+BET1P)*57.29578
      SAM2=(BETA2+BET2P)*57.29578
      SAM3=(BETA3+BET3P)*57.29578
      SAM4=(BETA4+BET4P)*57.29578
      IF(PARAM(062).EQ.0.0)GO TO 7312
      AM11=FAM1T(FR1,SAM1)
      AM12=FAM1T(FR2,SAM2)
      AM13=FAM1T(FR3,SAM3)
      AM14=FAM1T(FR4,SAM4)
      GO TO 7311
 7312 AM11=10.
      AM12=10.
      AM13=10.
      AM14=10.
 7311 CONTINUE
      AM11=AM11*SN1*0.025
      AM12=AM12*SN2*0.025
      AM13=AM13*SN3*0.025
      AM14=AM14*SN4*0.025
      IF(AM11.GT.29.9)AM11=29.9
      IF(AM12.GT.29.9)AM12=29.9
      IF(AM13.GT.29.9)AM13=29.9
      IF(AM14.GT.29.9)AM14=29.9
      U01=FU0I(SAM1,CV1)
      U02=FU0I(SAM2,CV2)
      U03=FU0I(SAM3,CV3)
      U04=FU0I(SAM4,CV4)
      U11=FU1I(SAM1)
      U12=FU1I(SAM2)
      U13=FU1I(SAM3)
      U14=FU1I(SAM4)
      U01=U01*SN1*0.025
      U02=U02*SN2*0.025
      U03=U03*SN3*0.025
      U04=U04*SN4*0.025
      U11=U11*SN1*0.025
      U12=U12*SN2*0.025
      U13=U13*SN3*0.025
      U14=U14*SN4*0.025
      AM21=U11-U01
      AM22=U12-U02
      AM23=U13-U03
      AM24=U14-U04
      SI1=U01/(AM11-AM21)
      SI2=U02/(AM12-AM22)
      SI3=U03/(AM13-AM23)
      SI4=U04/(AM14-AM24)
      BAM1=BETA1+BET1P
      BAM2=BETA2+BET2P
      BAM3=BETA3+BET3P
      BAM4=BETA4+BET4P
      XBM(1)=BAM1
      XBM(2)=BAM2
      XBM(3)=BAM3
      XBM(4)=BAM4
      SLP(1)=SLIP1
      SLP(2)=SLIP2
      SLP(3)=SLIP3
      SLP(4)=SLIP4
      CSI(1)=FCSI(SAM1,SLIP1)
      CSI(2)=FCSI(SAM2,SLIP2)
      CSI(3)=FCSI(SAM3,SLIP3)
      CSI(4)=FCSI(SAM4,SLIP4)
      F1=CSI(1)
      F2=CSI(2)
      F3=CSI(3)
      F4=CSI(4)
      GO TO 116
      F1=1.0
      F2=1.0
      F3=1.0
      F4=1.0
      TEMP=SQRT((CSI(1)*SLIP1)**2+BAM1**2)
      IF(TEMP.NE.0.0)F1=ABS(BAM1)/TEMP
      TEMP=SQRT((CSI(2)*SLIP2)**2+BAM2**2)
      IF(TEMP.NE.0.0)F2=ABS(BAM2)/TEMP
      TEMP=SQRT((CSI(3)*SLIP3)**2+BAM3**2)
      IF(TEMP.NE.0.0)F3=ABS(BAM3)/TEMP
      TEMP=SQRT((CSI(4)*SLIP4)**2+BAM4**2)
      IF(TEMP.NE.0.0)F4=ABS(BAM4)/TEMP
      IF((SLIP1.LE.0.).OR.(SLIP1.GE.PARAM(180)))GO TO 110
      IF(SLIP1.LT.(PARAM(180)/2.))GO TO 111
      F1=F1*(2.*(PARAM(180)-SLIP1)*(PARAM(181)-1.))/PARAM(180)+F1
      GO TO 110
  111 F1=F1*(2.*SLIP1*(PARAM(181)-1.))/PARAM(180)+F1
  110 IF((SLIP2.LE.0.).OR.(SLIP2.GE.PARAM(180)))GO TO 112
      IF(SLIP2.LT.(PARAM(180)/2.))GO TO 113
      F2=F2*(2.*(PARAM(180)-SLIP2)*(PARAM(181)-1.))/PARAM(180)+F2
      GO TO 112
  113 F2=F2*(2.*SLIP2*(PARAM(181)-1.))/PARAM(180)+F2
  112 IF((SLIP3.LE.0.).OR.(SLIP3.GE.PARAM(180)))GO TO 114
      IF(SLIP3.LT.(PARAM(180)/2.))GO TO 115
      F3=F3*(2.*(PARAM(180)-SLIP3)*(PARAM(181)-1.))/PARAM(180)+F3
      GO TO 114
  115 F3=F3*(2.*SLIP3*(PARAM(181)-1.))/PARAM(180)+F3
  114 IF((SLIP4.LE.0.).OR.(SLIP4.GE.PARAM(180)))GO TO 116
      IF(SLIP4.LT.(PARAM(180)/2.))GO TO 117
      F4=F4*(2.*(PARAM(180)-SLIP4)*(PARAM(181)-1.))/PARAM(180)+F4
      GO TO 116
  117 F4=F4*(2.*SLIP4*(PARAM(181)-1.))/PARAM(180)+F4
  116 CONTINUE
      FXU1=FR1*(THE-U1P*COSPS1-F1*AMU1*SINPS1*GR1)
      FXU2=FR2*(THE-U2P*COSPS2-F2*AMU2*SINPS2*GR2)
      FXU3=FR3*(THE-U3P*COSPS3-F3*AMU3*SINPS3*GR3)
      FXU4=FR4*(THE-U4P*COSPS4-F4*AMU4*SINPS4*GR4)
      G1=-PHI-U1P*SINPS1+F1*AMU1*COSPS1*GR1
      G2=-PHI-U2P*SINPS2+F2*AMU2*COSPS2*GR2
      G3=-PHI-U3P*SINPS3+F3*AMU3*COSPS3*GR3
      G4=-PHI-U4P*SINPS4+F4*AMU4*COSPS4*GR4
      FYU1=FR1*G1
      FYU2=FR2*G2
      FYU3=FR3*G3
      FYU4=FR4*G4
      FC1=FR1*U1P
      FC2=FR2*U2P
      FC3=FR3*U3P
      FC4=FR4*U4P
      FC1MAX=FR1*AM11*SI1
      FC2MAX=FR2*AM12*SI2
      FC3MAX=FR3*AM13*SI3
      FC4MAX=FR4*AM14*SI4
      SFXU=FXU1+FXU2+FXU3+FXU4
      SFYU=FYU1+FYU2+FYU3+FYU4
C     ZJUNK1=(ZR0-(ZR-Z1PP))/ALF
C     ZJUNK2=(ZR0-(ZR-Z2PP))/ALF
      ZJUNK1 =0.0
      ZJUNK2 =0.0
      SNPSIU=A*(FYU1+FYU2)-B*(FYU3+FYU4)+TF02*(FXU2-FXU1)
     1    +TR02*(FXU4-FXU3)
      IF(IAX.EQ.2)GO TO 4287
      SNTHEU=A*(S1P+S2P)-B*(S3P+S4P)-FXU1*Z1PP-FXU2*Z2PP
     1       -FXU3*Z3PP-FXU4*Z4PP
     1       -A*(FXU1*ZJUNK1+FXU2*ZJUNK2)-B*TANP*(FXU3+FXU4)
      SNPHIU=TF02*(S2P-S1P)+TS02*(S4P-S3P)+FYU1*(Z1PP+HFC)
     1       +FYU2*(Z2PP+HFC)+TLF*0.5*(FXU1*ZJUNK1-FXU2*ZJUNK2)
     1       -TS02*TANP*(FXU3-FXU4)-(FYU3+FYU4)*(CRRC*DEL3+HRC+ZR)
      GO TO 4288
 4287 SNTHEU=A*(S1P+S2P)-B*(S3P+S4P)-FXU1*Z1PP-FXU2*Z2PP
     1       -FXU3*Z3PP-FXU4*Z4PP
C    1       -FXU3*Z3PP-FXU4*Z4PP-A*(FXU1*ZJUNK1+FXU2*ZJUNK2)
C    1       -B*TANP*(FXU3+FXU4)+B*((FXU3*CPSR3+FYU3*SPSR3)*(ZR-Z3PP)
C    1         +(FXU4*CPSR4+FYU4*SPSR4)*(ZR-Z4PP))/TS02
      SNPHIU=TF02*(S2P-S1P)+TR02*(S4P-S3P)+FYU1*(Z1PP+HFC)
     1       +FYU2*(Z2PP+HFC)+FYU3*(Z3PP+HRC)+FYU4*(Z4PP+HRC)
C    1       +FYU2*(Z2PP+HFC)+TLF*0.5*(FXU1*ZJUNK1-FXU2*ZJUNK2)
C    1       -TS02*TANP*(FXU3-FXU4)+FYU3*(Z3PP+HRC)+FYU4*(Z4PP+HRC)
C    1       -TR*((FXU3*CPSR3+FYU3*SPSR3)*(ZR-Z3PP)
C    1         -(FXU4*CPSR4+FYU4*SPSR4)*(ZR-Z4PP))/TS
 4288 CONTINUE
      FS1=FR1*F1*AMU1*GR1
      FS2=FR2*F2*AMU2*GR2
      FS3=FR3*F3*AMU3*GR3
      FS4=FR4*F4*AMU4*GR4
      GO TO (1000,1000,3000),ISW
 1000 CONTINUE
      QDT=(SFXU/SM-SNTHEU/GAM2)*ONEOA
      UDT=V*R-W*Q-G*THE-(A2*SFXU/SM-A1*SNTHEU/GAM2)*ONEOA
      WDT=U*Q-V*P-(S1P+S2P+S3P+S4P)/AMS
      WDT=WDT+(FXU3+FXU4)*TANP/AMS
      TMP=W*P-U*R+G*PHI
      D1=SM*TMP+SFYU
      D2=-GAM3*TMP+SNPHIU
        D3=GAM1*TMP+SNPSIU
          VDT=(D1*F1+D2*GV1+D3*GV2)*ONEOD
      PDT=(-D1*F2+D2*GP1+D3*GP2)*ONEOD
      RDT=(D1*F3+D2*GR1+D3*GR2)*ONEOD
      PHIDT=P+R*THE
      THEDT=Q-R*PHI
      LIPSW=0
      IF(PARAM(176).EQ.2.)GO TO 101
      IF((PARAM(176).EQ.0.).AND.(TIME.GE.PARAM(177)))GO TO 101
      IF((PARAM(176).EQ.1.).AND.(PRSW.GT.0.))GO TO 101
      LIPSW=1
      PSIDT=0.
      YDT=0.
      GO TO 102
C 101 PSIDT=R
  101 PSIDT=R + Q*PHI
      XDT=U*CZ-V*SN
      YDT=U*SN+V*CZ
  102 CONTINUE
      IF(PARAM(193).NE.0.0)DELSWC=0.01745329*(PARAM(194)*Y+PARAM(195)*
     1    YDT)
      ZDT=W-U*THE+V*PHI
      DDT=CV1*PRSW
      ANUMDT=(FC1+FC2+FC3+FC4)
      ADENDT=(FC1MAX+FC2MAX+FC3MAX+FC4MAX)
      ISWPQ=PARAM(127)+0.5
      ISWPQ=ISWPQ+1
      GO TO (7001,7002,7003,7004),ISWPQ
      GO TO 7001
 7002 PDT=0.0
      GO TO 7001
 7003 QDT=0.0
      GO TO 7001
 7004 PDT=0.0
      QDT=0.0
 7001 CONTINUE
      GO TO (1100,1200),ISW
C     STOP 999
      WRITE(ITTY,99)
   99 FORMAT('0','STOP 999')
      READ(KEYBD,11000)LL
11000 FORMAT(I1)
 1100 U=U0+UDT*DT
      V=V0+VDT*DT
      W=W0+WDT*DT
      P=P0+PDT*DT
      Q=Q0+QDT*DT
      DD=DD0+DDT*DT
      R=R0+RDT*DT
      X=X0+XDT*DT
      Y=Y0+YDT*DT
      Z=Z0+ZDT*DT
      PHI=PHIO+PHIDT*DT
      THE=THEO+THEDT*DT
      IF(LIPSW.EQ.1)PSIO=-ATAN(V/U)
      PSI=PSIO+PSIDT*DT
      ANUM=ANUMO+ANUMDT*DT
      ADEN=ADENO+ADENDT*DT
      UDTO=UDT
      VDTO=VDT
      WDTO=WDT
      PDTO=PDT
      QDTO=QDT
      RDTO=RDT
      PHIDTO=PHIDT
      DDTO=DDT
      THEDTO=THEDT
      PSIDTO=PSIDT
      XDTO=XDT
      YDTO=YDT
      ZDTO=ZDT
      ANUMDO=ANUMDT
      ADENDO=ADENDT
      ISW=2
      GO TO 100
1200  TLT=0.5*DT
      V=VO+TLT*(VDT+VDTO)
      W=WO+TLT*(WDT+WDTO)
      P=PO+TLT*(PDT+PDTO)
      Q=QO+TLT*(QDT+QDTO)
      R=RO+TLT*(RDT+RDTO)
      X=XO+TLT*(XDT+XDTO)
      Y=YO+TLT*(YDT+YDTO)
      Z=ZO+TLT*(ZDT+ZDTO)
      DD=DDO+TLT*(DDT+DDTO)
      PHI=PHIO+TLT*(PHIDT+PHIDTO)
      THE=THEO+TLT*(THEDT+THEDTO)
      PSI=PSIO+TLT*(PSIDT+PSIDTO)
      U=UO+TLT*(UDT+UDTO)
      ANUM=ANUMO+TLT*(ANUMDT+ANUMDO)
      ADEN=ADENO+TLT*(ADENDT+ADENDO)
      PHIOUT=PHI+TLT*PHIDT
      THEOUT=THE+TLT*THEDT
      POUT=P+PDT*TLT
      QOUT=Q+QDT*TLT
      ROUT=R+RDT*TLT
      UOUT=U+UDT*TLT
      VOUT=V+VDT*TLT
      WOUT=W+WDT*TLT
      ZOUT=Z+ZDT*TLT
      XOUT=X+XDT*TLT
      YOUT=Y+YDT*TLT
      DOUT=DD+TLT*DDT
      PDTOUT=0.5*(3.0*PDT-PDTO)
      QDTOUT=0.5*(3.0*QDT-QDTO)
      RDTOUT=0.5*(3.0*RDT-RDTO)
      PSIOUT=PSI+TLT*PSIDT
      UDTOUT=0.5*(3.0*UDT-UDTO)
      VDTOUT=.5*(3.*VDT-VDTO)
      WDTOUT=0.5*(3.0*WDT-WDTO)
      ANOUT=ANUM+TLT*ANUMDT
      ADOUT=ADEN+TLT*ADENDT
      PO=P
      QO=Q
      DDO=DD
      RO=R
      UO=U
      VO=V
      WO=W
      XO=X
      YO=Y
      ZO=Z
      PHIO=PHI
      THEO=THE
      PSIO=PSI
      ANUMO=ANUM
      ADENO=ADEN
      P=POUT
      Q=QOUT
      R=ROUT
      DD=DOUT
      U=UOUT
      V=VOUT
      W=WOUT
      X=XOUT
      Y=YOUT
      Z=ZOUT
      PHI=PHIOUT
      THE=THEOUT
      PSI=PSIOUT
      ANUM=ANOUT
      ADEN=ADOUT
      ISW=3
      GO TO 100
 3000 P=PO
      Q=QO
      R=RO
      U=UO
      V=VO
      W=WO
      X=XO
      Y=YO
      Z=ZO
      PHI=PHIO
      DD=DDO
      THE=THEO
      PSI=PSIO
      THS1=POLY(DFL1,THEFNT)+THE1
      THS2=POLY(DFL2,THEFNT)+THE2
      PHS1=YHS1+PHI1
      PHS2=YHS2+PHI2
      IF(PARAM(193).NE.0.0)DFLSWC=0.01745329*(PARAM(194)*YOUT+PARAM(195)
     1      *YDT)
      CALL PTBAK(BETA1,FR1,AKK1,PTB1)
      CALL PTBAK(BETA2,FR2,AKK2,PTB2)
      IF(PARAM(60).NE.0.0)GO TO 4334
      PTB1=PTBR
      PTB2=PTBR
      AKK1=1.0
      AKK2=1.0
 4334 CONTINUE
      AMT1=FXU1*(PTB1*SINPS1-YSA1*COSPS1-Z1*(PHI1*COSPS1-PHS1))
     1    +FYU1*(-PTR1*AKK1*COSPS1-YSA1*SINPS1-Z1*(PHI1*SINPS1-THS1))   00019890
     1     -FR1*(-PTR1*(PHS1*COSPS1+THS1*SINPS1)+YSA1*(THS1*COSPS1-     00019900
     1       PHS1*SINPS1)-Z1*(PHS1*PHI1*SINPS1-THS1*PHI1*COSPS1))       00019910
      AMT2=FXU2*(PTR2*SINPS2-YSA2*COSPS2-Z2*(PHI2*COSPS2-PHS2))         00019920
     1    +FYU2*(-PTR2*AKK2*COSPS2-YSA2*SINPS2-Z2*(PHI2*SINPS2-THS2))   00019930
     1     -FR2*(-PTR2*(PHS2*COSPS2+THS2*SINPS2)+YSA2*(THS2*COSPS2-     00019940
     1       PHS2*SINPS2)-Z2*(PHS2*PHI2*SINPS2-THS2*PHI2*COSPS2))       00019950
      AMT1 = SWMT*AMT1                                                  00019960
      AMT2 = SWMT*AMT2                                                  00019970
      NNN=PARAM(198)/(PARAM(75)*UOUT)+0.5                               00019980
      XI(1)=XOUT+A*CZ-TF02*SN                                           00019990
      XI(2)=XOUT+A*CZ+TF02*SN                                           00020000
      XI(3)=XOUT-B*CZ-TR02*SN                                           00020010
      XI(4)=XOUT-B*CZ+TR02*SN                                           00020020
      NBMP=PARAM(277)+0.5                                               00020030
      DO 8498 I=1,4                                                     00020040
      DELX(I)=GETDEL(XI,I,PARAM(200),NBMP)                              00020050
 8498 CONTINUE                                                          00020060
      ETAX=(UDTOUT-VOUT*ROUT+WOUT*QOUT)/386.4                           00020070
      ETAL=(VDTOUT+ROUT*UOUT-WOUT*POUT)/386.4                           00020200
      AAK=AK1                                                           00020080
      IF(ETAX.GT.(-0.5))AAK=AK2                                         00020090
      FXL1=AML1*RRIM*ARPS1*ARPS1*WCTH1                                  00020100
      FXL2=AML2*RRIM*ARPS2*ARPS2*WCTH2                                  00020110
      FZL1=AML1*RRIM*ARPS1*ARPS1*WSTH1                                  00020120
      FZL2=AML2*RRIM*ARPS2*ARPS2*WSTH2                                  00020130
C ### BY-PASS DISPLAY DAC'S ###
C     IOUT(01)=POUT*5730.0+SIGN(0.5,POUT)                               00020140
C     IOUT(02)=ROUT*5730.0+SIGN(0.5,ROUT)                               00020150
C     IOUT(03)=PHIOUT*57300.0+SIGN(0.5,PHIOUT)                          00020160
C     IOUT(04)=THEOUT*57300.0+SIGN(0.5,THEOUT)                          00020170
C     IOUT(05)=PSIOUT*5730.0+SIGN(0.5,PSIOUT)                           00020180
C     IOUT(06)=ETAX*1000.0+SIGN(0.5,ETAX)                               00020190
C     IOUT(07)=ETAL*10000.0+SIGN(0.5,ETAL)                              00020210
C     IOUT(08)=DELSWC*1146.0+SIGN(0.5,DELSWC)                           00020220
      TEMP=AMT1/AIFW-RDTOUT                                             00020230
      TEMP=TEMP+(-FXL1*(YSA1+RWR/2.0))/AIFW                             00020240
C     IOUT(9)=TEMP*SFOUT(9)+SIGN(0.5,TEMP)                              00020250
      IOUT(01)=- TEMP*SFOUT(9)                                          0002025
      TEMP=RZF+ZOUT-A*THEOUT+TF02*PHIOUT                                00020260
      TEMP=TEMP-DELX(1)                                                 00020270
C     IOUT(10)=TEMP*SFOUT(10)+SIGN(0.5,TEMP)                            00020280
      IOUT(02)=TEMP*SFOUT(10)                                           00020280
      TEMP=AMT2/AIFW-RDTOUT                                             00020290
      TEMP=TEMP+(-FXL2*(YSA2+RWR/2.0))/AIFW                             00020300
C     IOUT(11)=TEMP*SFOUT(9)+SIGN(0.5,TEMP)                             00020310
      IOUT(03)=- TEMP*SFOUT(9)                                          0020310
      TEMP=RZF+ZOUT-A*THEOUT-TF02*PHIOUT                                00020320
      TEMP=TEMP-DELX(2)                                                 00020330
C     IOUT(12)=TEMP*SFOUT(10)+SIGN(0.5,TEMP)                            00020340
      IOUT(04)=TEMP*SFOUT(10)                                           00020340
C     TEMP=-(AM21-AM11)*0.05                                            00020350
      TEMP= (AM21-AM11)*0.05                                            00020350
C     IOUT(13)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020360
      IOUT(05)=TEMP                                                     00020360
      TM1=TF02*PDTOUT                                                   00020370
      TM2=A*QDTOUT                                                      00020380
      TM3=CIP                                                           00020390
      FRZ1=-FXU1*ZJUNK1                                                 00020400
      FRZ2=-FXU2*ZJUNK2                                                 00020410
```

```
C     TEMP=-TM1+TM2+TM3+2.0*FRZ1/AMUF                                   00020420
      TEMP=-TM1+TM2+TM3-FYU1*SPSR3
C     TEMP=TEMP+2.0*FZL1/AMUF                                           00020430
C     TEMP=TEMP+2.0*FZL2/AMUF                                           00020480
C     IOUT(14)=TEMP*SFOUT(14)+SIGN(0.5,TEMP)                            00020440
      IOUT(06)=TEMP*SFOUT(14)                                           00020440
C     TEMP=-(AM22-AM12)*0.05                                            00020450
      TEMP= (AM22-AM12)*0.05                                            00020450
C     IOUT(15)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020460
      IOUT(07)=TEMP                                                     00020460
C     TEMP= TM1+TM2+TM3+2.0*FRZ2/AMUF                                   00020470
      TEMP= TM1+TM2+TM3+FYU2*SPSR3
C     IOUT(16)=TEMP*SFOUT(14)+SIGN(0.5,TEMP)                            00020490
      IOUT(08)=TEMP*SFOUT(14)                                           00020490
C ### BY-PASS DISPLAY DAC'S ###
C     IOUT(17)=UOUT*5.6818+SIGN(0.5,UOUT)                               00020500
C     IOUT(18)=VOUT*50.0+SIGN(0.5,VOUT)                                 00020510
C     IOUT(19)=YOUT*4.166+SIGN(0.5,YOUT)                                00020520
C     IOUT(20)=-XOUT*0.8333+SIGN(0.5,-XOUT)                             00020530
C     IOUT(21)=-TQFBR*0.25+SIGN(0.5,TQFBR)                              00020540
      IOUT(09)=-TQFBR*0.25*SFOUT(14)                                    00020540
C     IOUT(22)=-TQRBR*0.25+SIGN(0.5,TQRBR)                              00020550
      IOUT(10)=-TQRBR*0.25*SFOUT(14)                                    0020550
C     IOUT(23)=DELSWC/10.0*T+SIGN(0.5,DELSWC)                           00020560
      IOUT(11)=DELSWC/10.0                                              00020560
      IOUT(12) = 0
      TEMP=(-TRO2+Z3*PHIOUT)                                            00020570
C     IOUT(25)=TEMP*SFOUT(25)+SIGN(0.5,TEMP)                            00020580
      IOUT(13)=TEMP*SFOUT(25)                                           00020580
      TEMP=CIVP-B*QDTOUT                                                00020590
      TEMP=TEMP+(FXU3+FXU4)*TANP/AMUR                                   00020600
C     IOUT(26)=TEMP*SFOUT(14)+SIGN(0.5,TEMP)                            00020610
      IOUT(14)=      TEMP*SFOUT(14)                                     00020610
      TEMP=(TRO2+Z4*PHIOUT)                                             00020620
C     IOUT(27)=TEMP*SFOUT(27)+SIGN(0.5,TEMP)                            00020630
      IOUT(15)=TEMP*SFOUT(27)                                           00020630
      TEMP=-PDTOUT-(FYU3*(-TRO2*PHIOUT-Z3)-FYU4*(TRO2*PHIOUT-Z4))/AIR   00020640
      TEMP=TEMP-((FYU3+FYU4)*(CRRC*DEL3+HRC)+TSO2*(FXU3-FXU4)*TANP)/AIR 00020650
C     IOUT(28)=TEMP*SFOUT(28)+SIGN(0.5,TEMP)                            00020660
      IOUT(16)=TEMP*SFOUT(28)                                           00020660
C     TEMP=-(AM23-AM13)*0.05                                            00020670
      TEMP= (AM23-AM13)*0.05                                            00020670
C     IOUT(29)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020680
      IOUT(17)=TEMP                                                     00020680
      TEMP=(RZR+ZOUT+B*THEOUT+PHIOUT*TRO2)                              00020690
      TEMP=TEMP-DELX(3)                                                 00020700
C     IOUT(30)=TEMP*SFOUT(10)+SIGN(0.5,TEMP)                            00020710
      IOUT(18)=TEMP*SFOUT(10)                                           00020710
C     TEMP=-(AM24-AM14)*0.05                                            00020720
      TEMP= (AM24-AM14)*0.05                                            00020720
C     IOUT(31)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020730
      IOUT(19)=TEMP                                                     0002073
      TEMP=(RZR+ZOUT+B*THEOUT-PHIOUT*TRO2)                              00020740
      TEMP=TEMP-DELX(4)                                                 00020750
C     IOUT(32)=TEMP*SFOUT(10)+SIGN(0.5,TEMP)                            00020760
      IOUT(20)=TEMP*SFOUT(10)                                           00020760
      TEMP=U01*0.05                                                     00020770
C     IOUT(33)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020780
      IOUT(21)=TEMP                                                     00020780
      TEMP=U02*0.05                                                     00020790
```

```
C     IOUT(34)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020800
      IOUT(22)=TEMP                                                     00020800
      TEMP=AM11*0.05                                                    00020810
C     IOUT(35)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020820
      IOUT(23)=TEMP                                                     00020820
      TEMP=AM12*0.05                                                    00020830
C     IOUT(36)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020840
      IOUT(29)=TEMP                                                     00020840
C     IOUT(37)=UG1P*SFOUT(37)+SIGN(0.5,UG1P)                            00020850
      IOUT(25)=UG1P*SFOUT(37)                                           00020850
C     IOUT(38)=UG2P*SFOUT(37)+SIGN(0.5,UG2P)                            00020860
      IOUT(26)=UG2P*SFOUT(37)                                           00020860
C     IOUT(39)=UG3P*SFOUT(37)+SIGN(0.5,UG3P)                            00020870
      IOUT(27)=UG3P*SFOUT(37)                                           00020870
C     IOUT(40)=UG4P*SFOUT(37)+SIGN(0.5,UG4P)                            00020880
      IOUT(24)=UG4P*SFOUT(37)                                           00020880
      TEMP=U03*0.05                                                     00020890
C     IOUT(41)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020900
      IOUT(28)=TEMP                                                     00020900
      TEMP=U04*0.05                                                     00020910
C     IOUT(42)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020920
      IOUT(30)=TEMP                                                     0     9
      TEMP=AM13*0.05                                                    00020930
C     IOUT(43)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020940
      IOUT(31)=TEMP                                                     00020940
      TEMP=AM14*0.05                                                    00020950
C     IOUT(44)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020960
      IOUT(32)=TEMP                                                     00020960
C     TEMP=SI1                                                          00020970
      TEMP=-SI1                                                          0002097
C     IOUT(45)=TEMP*10000.0+SIGN(0.5,TEMP)                              00020980
      IOUT(33)=TEMP                                                     00020980
C     TEMP=SI2                                                          00020990
      TEMP=-SI2                                                         00020990
C     IOUT(46)=TEMP*10000.0+SIGN(0.5,TEMP)                              00021000
      IOUT(34)=TEMP                                                     00021000
C     TEMP=SI3                                                          00021010
      TEMP=-SI3                                                          0002101
C     IOUT(47)=TEMP*10000.0+SIGN(0.5,TEMP)                              00021020
      IOUT(35)=TEMP                                                     00021020
C     TEMP=SI4                                                          00021030
      TEMP=-SI4                                                          0002103
C     IOUT(48)=TEMP*10000.0+SIGN(0.5,TEMP)                              00021040
      IOUT(36)=TEMP                                                     00021040
      IOUT(37)=0
      IF(IAX.EQ.1)GO TO 7719                                            00021050
C     TEMP=G*(1.0+A*AMS/(AMUR*(A+B)))-B*QDTOUT-TRO2*PDTOUT+             00021060
C    1   2.0*(FXU3*CPSR3+FYU3*SPSR3)*(ZR-Z3PP)/(AMUR*TS)                00021070
      TEMP    =G*(1.+A*AMS/(AMUR*(A+B)))-B*QDTOUT-TRO2*PDTOUT
     1          -FYU3*SPSR4
C     TEMP    =G*(1.+A*AMS/(AMUR*(A+B)))-B*QDTOUT-TRO2*PDTOUT
C    1          -FYU3*SPSR4
C     IOUT(24)=TEMP+SIGN(0.5,TEMP)                                      00021080
      IOUT(12)=TEMP*SFOUT(14)                                            0002108
C     TEMP=G*(1.0+A*AMS/(AMUR*(A+B)))-B*QDTOUT+TRO2*PDTOUT+             00021090
C    1   2.0*(FXU4*CPSR4+FYU4*SPSR4)*(ZR-Z4PP)/(AMUR*TS)                00021100
      TEMP    =G*(1.+A*AMS/(AMUR*(A+B)))-B*QDTOUT+TRO2*PDTOUT
     1          +FYU4*SPSR4
C     TEMP    =G*(1.+A*AMS/(AMUR*(A+B)))-B*QDTOUT+TRO2*PDTOUT
C    1          +FYU4*SPSR4
```

```
C     IOUT(28)=TEMP+SIGN(0.5,TEMP)
      IOUT(37)=TEMP*SFOUT(14)
C     IOUT(25)=0
      IOUT(13)=0
C     IOUT(27)=0
C     IOUT(15)=0
      IOUT(16)=0
 7719 CONTINUE
      IF(UOUT.GE.25.0*5280.*12.0/3600.0)TIME25=TIME
      IF(UOUT.GE.10.0*5280.*12.0/3600.0)TIME10=TIME
      IF(UOUT.GE.5.0*5280.0*12.0/3600.0 )TIME5=TIME
      TEMPE=ABS(ROUT*57.29578)
      IF(TEMPE.GT.RMAX)RMAX=TEMPE
      TEMPE=ABS(ETAL)
      IF(TEMPE.GT.ETAMAX)ETAMAX=TEMPE
      TEMPE=ABS(PHIOUT*57.29578)
      IF(TEMPE.GT.PHIMAX)PHIMAX=TEMPE
      TEMPE=ABS(DELSWC*57.29578)
      IF(TEMPE.GT.DSWMAX)DSWMAX=TEMPE
      IF(TIME.LE.5.0)PSI5=PSIOUT*57.29578
      INTEG=PARAM(128)+0.5
      ITEMP=INTEG+1
      BTV=VOUT/UOUT
      BTVDT=(UOUT*VDTOUT-VOUT*UDTOUT)/(UOUT*UOUT)
      ONER=(ROUT+BTVDT)/SQRT(UOUT**2+VOUT**2)
C     ONER=ONER*20.0
      TEMPE=ABS(BTV)
      IF(TEMPE.GT.BTVMAX)BTVMAX=TEMPE
C###################################
C#####   APL DISPLAY PROGRAM   #####
C#####################################################
C###################   IREC =1: AX,AY,R,U,V,B,1/R   ###
C#####################################################
      IREC   =PARAM(43)
      GO TO(1,2,3,4),IREC
    1 CONTINUE
      IOUT(13)=ETAX/5.
      IOUT(14)=ETAL/5.
      IOUT(15)=ROUT
      IOUT(16)=UOUT/1200.
      IOUT(38)=VOUT/1200.
      IOUT(39)=BTV/3.14
      IOUT(40)=ONER*240.
      GO TO 6123
    2 CONTINUE
C###############################################
C#####   SECOND DISPLAY GROUP ADDED HERE   #####
C###############################################
      IOUT(13) = POUT
      IOUT(14) = QOUT
      IOUT(15) = PHIOUT * 57.3 / 10.
      IOUT(16) = THEOUT * 57.3 / 10.
      IOUT(38) = PSIOUT * 57.3 / 10.
      IOUT(39) = PE/1000.
      IOUT(40) = WOUT / 1200.
      GO TO 6123
    3 CONTINUE
      GO TO 6123
    4 CONTINUE
      GO TO 6123
      GO TO (6123,6124,6125,6126,6127,6128,6129,6130,6131,6132,6133,     00021350
     1    6134,6135,6136,6137,6138,6139,6140,6141),ITEMP                  00021360
 6124 IOUT(07)=PF*T/1000.+SIGN(0.5,PF)                                    00021370
      IOUT(08)=THEOUT*10000.0*57.3/10.0+SIGN(0.5,THEOUT)                  00021380
      IOUT(17)=QOUT*5730.0+SIGN(0.5,QOUT)                                 00021390
      IOUT(18)=SNTHEU*10000.0/10000.0+SIGN(0.5,SNTHEU)                    00021400
      GO TO 3791                                                          00021410
 6125 IOUT(07)=FS1*10000.0/2500.0+SIGN(0.5,FS1)                           00021420
      IOUT(08)=FS2*10000.0/2500.0+SIGN(0.5,FS2)                           00021430
      IOUT(17)=FS3*10000.0/2500.0+SIGN(0.5,FS3)                           00021440
      IOUT(18)=FS4*10000.0/2500.0+SIGN(0.5,FS4)                           00021450
      GO TO 3791                                                          00021460
 6126 IOUT(07)=PF*T/1000.0+SIGN(0.5,PF)                                   00021470
      GO TO 3791                                                          00021480
 6127 IOUT(07)=FS1*10000.0/2500.0+SIGN(0.5,FS1)                           00021490
      IOUT(08)=FS2*10000.0/2500.0+SIGN(0.5,FS2)                           00021500
      IOUT(17)=FS3*10000.0/2500.0+SIGN(0.5,FS3)                           00021510
      IOUT(18)=FS4*10000.0/2500.0+SIGN(0.5,FS4)                           00021520
      GO TO 3791                                                          00021530
 6128 IOUT(17)=PSIOUT*5730.0+SIGN(0.5,PSIOUT)                             00021540
      IOUT(18)=POUT*5730.0+SIGN(0.5,POUT)                                 00021550
      GO TO 3791                                                          00021560
 6129 IOUT(07)=PF*T/1000.0+SIGN(0.5,PF)                                   00021570
      IOUT(08)=PHIOUT*57300.0+SIGN(0.5,PHIOUT)                            00021580
      IOUT(17)=PSIOUT*5730.0+SIGN(0.5,PSIOUT)                             00021590
      IOUT(18)=POUT*5730.0+SIGN(0.5,POUT)                                 00021600
 3791 IOUT(01)=ETAX*10000.0/2.5+SIGN(0.5,ETAX)                            00021610
      IOUT(02)=ETAL*10000.0+SIGN(0.5,ETAL)                                00021620
      IOUT(03)=ROUT*10000.0*57.3/100.+SIGN(0.5,ROUT)                      00021630
      IOUT(04)=CV1*100.0+SIGN(0.5,CV1)                                    00021640
      IOUT(05)=BTV*10000.0*57.3/20.0+SIGN(0.5,BTV)                        00021650
      IOUT(06)=ONER*10000.0+SIGN(0.5,ONER)                                00021660
      GO TO 6123                                                          00021670
 6130 IOUT(05)=S1P*10000.0/2000.0+SIGN(0.5,S1P)                           00021680
      IOUT(03)=S2P*10000.0/2000.0+SIGN(0.5,S2P)                           00021690
      IOUT(04)=S3P*10000.0/2000.0+SIGN(0.5,S3P)                           00021700
      IOUT(02)=S4P*10000.0/2000.0+SIGN(0.5,S4P)                           00021710
      GO TO 6123                                                          00021720
 6131 TEMP=AMU1                                                           00021730
      IOUT(05)=TEMP*10000.0+SIGN(0.5,TEMP)                                00021740
      TEMP=AMU2                                                           00021750
      IOUT(03)=TEMP*10000.0+SIGN(0.5,TEMP)                                00021760
      TEMP=AMU3                                                           00021770
      IOUT(04)=TEMP*10000.0+SIGN(0.5,TEMP)                                00021780
      TEMP=AMU4                                                           00021790
      IOUT(02)=TEMP*10000.0+SIGN(0.5,TEMP)                                00021800
      GO TO 6123                                                          00021810
 6132 IOUT(05)=PSI1*10000.0*57.3/10.0+SIGN(0.5,PSI1)                      00021820
      IOUT(03)=PSI2*10000.0*57.3/10.0+SIGN(0.5,PSI2)                      00021830
      IOUT(04)=PSI3*10000.0*57.3/10.0+SIGN(0.5,PSI3)                      00021840
      IOUT(02)=PSI4*10000.0*57.3/10.0+SIGN(0.5,PSI4)                      00021850
      GO TO 6123                                                          00021860
 6133 IOUT(05)=DEL1*10000.0/5.0+SIGN(0.5,DEL1)                            00021870
      IOUT(03)=DEL2*10000.0/5.0+SIGN(0.5,DEL2)                            00021880
      IOUT(04)=DEL3*10000.0/5.0+SIGN(0.5,DEL3)                            00021890
      IOUT(02)=PHIR*10000.0*57.32/10.0+SIGN(0.5,PHIR)                     00021900
      GO TO 6123                                                          00021910
 6134 IOUT(5)=-0.8333*XOUT+SIGN(0.5,-XOUT)                                00021920
      GO TO 6123                                                          00021930
 6135 IOUT(5)=10000.*F1 +SIGN(0.5,F1)                                     00021940
```

```
      IOUT(03)=10000.*F2+SIGN(0.5,F2)
      IOUT(4)=10000.*F3+SIGN(0.5,F3)
      IOUT(02)=10000.*F4+SIGN(0.5,F4)
      GO TO 6123
 6136 IOUT(5)=10000.*U1P    +SIGN(0.5,U1P)
      IOUT(03)=10000.*U2P  +SIGN(0.5,U2P)
      IOUT(04)=10000.*U3P     +SIGN(0.5,U3P)
      IOUT(02)=10000.*U4P     +SIGN(0.5,U4P)
      GO TO 6123
 6137 IOUT(05)=100.*DEL1DT    +SIGN(0.5,DEL1DT)
      IOUT(03)=100.*DEL2DT    +SIGN(0.5,DEL2DT)
      IOUT(04)=100.*DEL3DT       +SIGN(0.5,DEL3DT)
      IOUT(02)=40000.*PHIRD   +SIGN(0.5,PHIRD)
      GO TO 6123
 6138 IOUT(5 )=10000.*QUAN1   +SIGN(0.5,QUAN1)
      IOUT(03)=10000.*QUAN2    +SIGN(0.5,QUAN2)
      IOUT(04)=10000.*QUAN3     +SIGN(0.5,QUAN3)
      IOUT(02)=10000.*QUAN4    +SIGN(0.5,QUAN4)
      GO TO 6123
 6139 TEMP=RZR+ZOUT
      IOUT(30)=TEMP+SIGN(0.5,TEMP)
      IOUT(32)=TEMP+SIGN(0.5,TEMP)
      GO TO 6123
 6140 IOUT(05)=FYU1*10000.0/5000.0+SIGN(0.5,FYU1)
      IOUT(03)=FYU2*10000.0/5000.0+SIGN(0.5,FYU2)
      IOUT(04)=FYU3*10000.0/5000.0+SIGN(0.5,FYU3)
      IOUT(02)=FYU4*10000.0/5000.0+SIGN(0.5,FYU4)
      GO TO 6123
 6141 T123=10000.0/250000.0
      IOUT(03)=SNTHEU*T123+SIGN(0.5,SNTHEU)
      IOUT(05)=SFYU*10000.0/5000.0+SIGN(0.5,SFYU)
      IOUT(04)=SNPHIU*T123+SIGN(0.5,SNPHIU)
      IOUT(02)=SNPSIU*T123+SIGN(0.5,SNPSIU)
 6123 CONTINUE
      IF(INTFUN.NE.0)GO TO 3695
      TSWT=0.0
      IF(Y3.GT.5.0.OR.Y3.LT.0.0)TSWT=1.0
      IF(TSWT.NE.0.0.AND.TOO.EQ.0.0)TOO=XOUT
 3695 CONTINUE
      NUM=PARAM(044)+0.5
      IF(PARAM(190).NE.0.0.AND.NUM/2*2.EQ.NUM)GO TO 8432
      YSPEC=YOUT
      XPF=XOUT
      GO TO 7521
 8432 IF(XOUT.LE.XPF)YSPEC=YOUT
      IF(XOUT.LE.XPF)XEND=TIME+1.0
 7521 CONTINUE
 5151 CONTINUE
      IF(ABS(PSIOUT).GT.ABS(PSIM))PSIM=PSIOUT
      IF(TIME.NE.0.0)GO TO 795
      DO 6240 I=1,48
      ITMP(I)=IOUT(I)
 6240 CONTINUE
  795 TIME=TIME+DT
      TEMPE=ABS(TQRBR)
      IF(TEMPE.GT.TQRMAX)TQRMAX=TEMPE
      TEMPE=ABS(TQFBR)
      IF(TEMPE.GT.TQFMAX)TQFMAX=TEMPE
      TEMPE=0.0
      IF(DELSWC.NE.0.0)TEMPE      =(ROUT*(A+B)*ANG*(ANL1+ANL2))/
```

```
     1    (2.0*UOUT*DELSWC)                                                      00022550
      TEMPE=ABS(TEMPE)                                                           00022560
      IF(TEMPE.GT.TMAX1)TMAX1=TEMPE                                              00022570
      TEMPE=0.0                                                                  00022580
      IF(DELSWC.NE.0.0)TEMPE       =ROUT*ANG*(ANL1+ANL2)/(2.0*DELSWC)            00022590
      TEMPE=ABS(TEMPE)                                                           00022600
      IF(TEMPE.GT.TMAX2)TMAX2=TEMPE                                              00022610
      TEMPE      =2.0*DELSWC*UOUT*UOUT/(G*(A+B)*ANG*(ANL1+ANL2))                 00022620
      TEMPE=ABS(TEMPE)                                                           00022630
      IF(TEMPE.GT.TMAX3)TMAX3=TEMPE                                              00022640
      TEMPE=ABS(PSIOUT)                                                          00022650
      IF(TEMPE.GT.PSIMAX)PSIMAX=TEMPE                                            00022660
C 997 FORMAT('0'8I15/(' '8I15))                                                  00022670
  997 FORMAT('0',8I15/(' ',8I15))                                                00022670
      NUMBR = NUMBR + 1                                                          00022680
C     IF(IPRT.NE.0)PRINT 997,(IOUT(I),I=1,48)                                    00022690
      IF(IPRT.NE.0)WRITE(ITTY,997)(IOUT(I),I=1,48)                               00022690
C        LI,7       X'4000'                                                      00022700
C      WD,7 X'1102'                                                              00022710
      RETURN                                                                     00022720
      END                                                                        00022730
```

```
C     SUBROUTINE SBPG4                                                  00023359
      SUBROUTINE SBPG4                                                  00023360
      DIMENSION NAMEX(124),NAME(289)
C     DIMENSION NAME(289)                                               00023370
      COMMON/API/ OPEN  ,RTSW  ,LDTSW ,RBSW
      COMMON/DEVICE/KEYBD,ITTY,ICDRD,LPTR
      COMMON/SPLTAX/SPSR3,CPSR3,SPSR4,CPSR4,SCR3,SCR4,TBCR3,TBCR4,TBSR3,00023380
     1   TBSR4,TRSR3,TRSR4,TRCR3,TRCR4,PSR3,PSR4,IAX                    00023390
      COMMON/AFU1/AF11,AF12,AF13,GAMC1                                  00023400
      COMMON/AFU0/AF01,AF02,AF03,AF04,AF05,AF06,GAMC0                   00023410
      COMMON/PTBK/AP1,AP2,AP3,AP4,AP5,BTC1,BTC2                         00023420
      COMMON/EFFS/EFF1,EFF2,THE1,THE2                                   00023430
      COMMON/FRIDAY/B1VMAX,BTVTB(100)                                   00023440
      COMMON/THINGS/TMAX1,TMAX2,TMAX3,TQRMAX,TQFMAX,PSIMAX,ONER
      COMMON/NEWER/TIME25,TIME10,RMAX,ETAMAX,PSI5,PHIMAX,DSWMAX         00023460
      COMMON/DDD/DIN,DDTO,DDT,TIME5,DDO,DD,DOUT                         00023470
      COMMON/EES/O1,O2,O3,E4,E5,E6                                      00023480
      COMMON/ALPHA/ALPH(20)                                             00023490
      COMMON/COMBLK/AIXP,SM,AIYP,AIXZP,GAM1,GAM2,GAM3,AIXBR,AIYBR,      00023500
     1     AIZBR,A1,A2,AIXZBR,A12,E1,E2,E3,DELTA,GV1,GV2,GP1,GP2,GR1,   00023510
     1     GR2,CIP,CIVP,RZF,RZR,A2T,CA20,CA23,TIME,DT,      ANGNL,ANGNLO  00023520
     1     ,TRO2,TFO2,TSO2,G,THRD,TWN7                                  00023530
      COMMON/EFFS/ANUM,ADEN,ANUMDT,ADENDT,ANUMO,ADENO,ANUMDO,ADENDO,    00023540
     1  ANOUT,ADOUT                                                     00023550
      COMMON/XX/IDAC(48),IADC(32)                                       00023560
      COMMON/INOUT/INA(32),IOUTA(48),IN(32),IOUT(48),ISW1,ISW7,SFIN(32),00023570
     1      SFOUT(48),IPRT,ITMP(48)                                     00023580
      COMMON/UVW/VC,UIN                                                 00023590
      COMMON/ANALOG/DEL1DT,DEL2DT,DEL3DT,DEL1,DEL2,DEL3,PHIRD,PHIR,     00023600
     1    DELEW1,DELEW2,U1P,U2P,U3P,U4P,S1P,S2P,S3P,S4P,TTA,PFSLP       00023610
      COMMON/OPSW/IHSW                                                  00023620
      COMMON/QUANTS/QUAN1,QUAN2,QUAN3,QUAN4                             00023630
      COMMON/VARS/P,Q,R,U,V,W,X,Y,Z,THE,PHI,PSI,PO,QO,RO,UO,VO,WO,XO,   00023640
     1    YO,ZO,THEO,PHIO,PSIO                                          00023650
      COMMON/SPZBLK/N1,N2,IPOT(244),IPOTAD(244),PARAM(300)              00023660
      COMMON/MICKEY/IRUNTB(100),VTB(100),SNLTB(100),SNRTB(100),DSWTB(10000023670
     1 ),BTTB(100),DTB(100),EFFTB(100),DYTB(100),PSITB (100),NRUN,      00023680
     1   YSPEC,PSIM,XPF                                                 00023690
     1   ,TMPTB(100)                                                    00023700
      COMMON/XBS/XB(12),NS(4,12),DELX(4),XI(4),NNN                      00023710
      COMMON/NONAME/XEND,O,EXIT2
      EQUIVALENCE                                                       00023730
     1   (PARAM( 1),AMS)     , (PARAM( 2),AMUF)  , (PARAM( 3),AMUR)   , 00023740
     1   (PARAM( 4),ZF)      , (PARAM( 5),ZR)    , (PARAM( 6),A)      , 00023750
     1   (PARAM( 7),B)       , (PARAM( 8),TF)    , (PARAM( 9),TR)     , 00023760
     1   (PARAM(10),TS)      , (PARAM(11),AIX)   , (PARAM(12),AIY)    , 00023770
     1   (PARAM(13),AIZ)     , (PARAM(14),AIXZ)  , (PARAM(15),AIR)    , 00023780
     1   (PARAM(16),CF)      , (PARAM(17),RF)    , (PARAM(18),CFP)    , 00023790
     1   (PARAM(19),AKF)     , (PARAM(20),ALAMF) , (PARAM(21),OFC)    , 00023800
     1   (PARAM(22),OFT)     , (PARAM(23),CR)    , (PARAM(24),RR)     , 00023810
     1   (PARAM(25),CRP)     , (PARAM(26),AKR)   , (PARAM(27),ALAMR)  , 00023820
     1   (PARAM(28),ORC)     , (PARAM(29),ORT)   , (PARAM(30),AKRS)   , 00023830
     1   (PARAM(31),RW)      , (PARAM(32),B4)    , (PARAM(33),OT)     , 00023840
     1   (PARAM(34),CA0)     , (PARAM(35),CA1)   , (PARAM(36),CA2)    , 00023850
     1   (PARAM(37),CA3)     , (PARAM(38),CA4)   , (PARAM(39),AISW)   , 00023860
     1   (PARAM(44),AKDL)    , (PARAM(41),AKSC)  , (PARAM(42),ANG)    , 00023870
     1   (PARAM(43),WG)      , (PARAM(40),ANL2)  , (PARAM(45),AKSL)     00023880
      EQUIVALENCE                                                       00023890
     1   (PARAM(46),ANL1)    , (PARAM(47),AIFW)  , (PARAM(48),HDL)    , 00023900
```

```
1    (PARAM(49),AIWF)    , (PARAM(50),AIWR)   , (PARAM(51),AID)     ,  00023910
1    (PARAM(52),ARBR)    , (PARAM(53),EPS1)   , (PARAM(54),EPS2)    ,  00023920
1    (PARAM(55),PTBR)    , (PARAM(56),YSA1)   , (PARAM(57),YSA2)    ,  00023930
1    (PARAM(58),YHS1)    , (PARAM(59),YHS2)   , (PARAM(60),AKD)     ,  00023940
1    (PARAM(61),TQDBR)   , (PARAM(62),AK)     , (PARAM(63),PIN)     ,  00023950
1    (PARAM(64),QIN)     , (PARAM(65),RIN)    , (PARAM(66),UIZ)     ,  00023960
1    (PARAM(67),VIN)     , (PARAM(68),WIN)    , (PARAM(69),XIN)     ,  00023970
1    (PARAM(70),YIN)     , (PARAM(71),ZIN)    , (PARAM(72),THEIN)   ,  00023980
1    (PARAM(73),PHIIN)   , (PARAM(74),PSIIN)  , (PARAM(75),DTIN)    ,  00023990
1    (PARAM(76),TEND)    , (PARAM(77),AKT1)   , (PARAM(78),AKT2)  ,    00024000
1    (PARAM(79),AKT3)    , (PARAM(80),AKT4)   , (PARAM(81),RPS1)  ,    00024010
1    (PARAM(82),RPS2)    , (PARAM(83),RPS3)   , (PARAM(84),RPS4)    ,  00024020
1    (PARAM(85),B1)      , (PARAM(86),B2)     , (PARAM(87),B3)         00024030
 EQUIVALENCE                                                           00024040
1                        (PARAM( 89),DEL1DN),(PARAM( 90),DEL2DN),      00024050
1    (PARAM( 91),DEL3DN),(PARAM( 92),DEL1IN), (PARAM( 93),DEL2IN),     00024060
1    (PARAM( 94),DEL3IN), (PARAM( 95),PHIDN), (PARAM(96),PHIRN),       00024070
1    (PARAM( 97),DEW1IN), (PARAM( 98),DEW2IN),(PARAM( 99),U1PIN),      00024080
1    (PARAM(100),U2PIN),  (PARAM(101),U3PIN), (PARAM(102),U4PIN),      00024090
1    (PARAM(103),S1PIN),  (PARAM(104),S2PIN), (PARAM(105),S3PIN),      00024100
1    (PARAM(106),S4PIN), (PARAM(107),PPRT)                             00024110
1,(PARAM(110),TQMAX),(PARAM(111),AKTQ),(PARAM(112),VCIN)               00024120
1,(PARAM(113),SWMT),(PARAM(114),DSWCM),(PARAM(115),TST),               00024130
1(PARAM(116),DSLP),(PARAM(117),CGAM),(PARAM(118),CS)                   00024140
1  ,(PARAM(119),TQRBR),(PARAM(120),TQFBR)                              00024150
1  ,(PARAM(121),PFL),(PARAM(122),TTD),(PARAM(123),DSW)                 00024160
1    ,(PARAM(124),TSW)                                                 00024170
 EQUIVALENCE                                                           00024180
1   (PARAM(130),AMCR),(PARAM(131),ESP),(PARAM(132),AKSL1),             00024190
1   (PARAM(133),AKSL2),(PARAM(134),AA1),(PARAM(135),AA2),              00024200
1   (PARAM(136),CCR),(PARAM(137),CFCR),(PARAM(138),AP),                00024210
1   (PARAM(139),EP1),(PARAM(140),EP2),(PARAM(141),ERR1),               00024220
1   (PARAM(142),ERR2),                                                 00024230
1   (PARAM(143),AML1),(PARAM(144),AML2),(PARAM(145),RRIM),             00024240
1    (PARAM(146),RWR),(PARAM(147),AKV),                                00024250
1     (PARAM(148),AK1  ) , (PARAM(149),AK2 ) , (PARAM(150),AM   ) ,    00024260
1     (PARAM(151),PM   ) , (PARAM(152),PV  ) , (PARAM(153),RT   ) ,    00024270
1     (PARAM(154),TD1  ) , (PARAM(155),TD2 ) , (PARAM(156),TD3  ) ,    00024280
1     (PARAM(157),TD4  ) , (PARAM(158),Y1  ) , (PARAM(159),YMAX) ,     00024290
1     (PARAM(160),ZET1) , (PARAM(161),ZET2) , (PARAM(162),EX1  ) ,     00024300
1     (PARAM(163),TBF  ) , (PARAM(164),TBR ) , (PARAM(165),T1   ) ,    00024310
1     (PARAM(166),WN1  ) , (PARAM(167),WN2 ) , (PARAM(168),R1   ) ,    00024320
1     (PARAM(169),PA0  )                                               00024330
1    ,(PARAM(196),EPSK1),(PARAM(197),EPSK2)                            00024340
1  ,(PARAM(274),TLF),(PARAM(275),ZR0),(PARAM(276),ALF)                 00024350
 EQUIVALENCE                                                           00024360
1    (PARAM(281),TSF),(PARAM(282),TSR),(PARAM(283),TANP),              00024370
1    (PARAM(284),HFC),(PARAM(285),HRC),(PARAM(286),CRRC)               00024380
 EQUIVALENCE
1    (PARAM(290),ROT),(PARAM(291),RA0),(PARAM(292),RA1),
1    (PARAM(293),RA2),(PARAM(294),RA3),(PARAM(295),RA4)
 EQUIVALENCE (NAME(172),NAMEX(1))
 EQUIVALENCE (PHIRD,DEL4DT),(PHIR,DEL4)
 DATA NAME/'  MS','  MUF',' MUR','  ZF','  ZR','   A','   B','  TF',00024390
1    '  TR','  TS','  IX','  IY','  IZ',' IXZ','  IR','  CF','  RF',00024400
1    'CFPR','  KF','LAMF','OMFC','OMFT','  CR','  RR','CRPR','  KR',00024410
1    'LAMR','OMRC','OMRT',' KRS','  RW','  R4',' OMT','  A0','  A1',00024420
1    '  A2','  A3','  A4',' ISW','ANL2',' KSC','  NG','  WG',' RUN',00024430
1    ' KSL','ANL1',' IFW','HDEL',' IWF',' IWR','  ID','  AR','EPS1',00024440
1    'EPS2','  PT','YSA1','YSA2','PHS1','PHS2','  KD',' TQD','   K',00024450
```

```
     1    'P-IN','Q-IN','R-IN','U-IN','V-IN','W-IN','X-IN','Y-IN','Z-IN',00024460
     1    'THIN','PHIN','PSIN','  DT','  TN',' KT1',' KT2',' KT3',' KT4',00024470
     1    'RPS1','RPS2','RPS3','RPS4','  B1','  B2','  B3','ITSC','D1DT',00024480
     1    'D2DT','D3DT','DEL1','DEL2','DEL3','PHDT','PHIR','DFW1','DFW2',00024490
     1    'U1PR','U2PR','U3PR','U4PR','S1PR','S2PR','S3PR','S4PR','PPRT',00024500
     1    'IPRT','TSTP','TQMX',' KTQ','  VC','MTSW','DSWM',' TST','DSLP',00024510
     1    'CGAM','  CS',' TQR',' TQF',' PFL','  T1',' DSW',' TSW','ISW5',00024520
     1    'SW15',                                                        00024530
     1    '    ','    ','    ','AMCR',' FSP','KSL1','KSL2',' AA1',' AA2',00024540
     1    ' CCR','CFCR','  AP',' FP1',' FP2','ERR1','ERR2','AML1','AML2',00024550
     1    'RRIM',' RWR','  KV','  K1','  K2','   M','  PM','  PV','  RT',00024560
     1    ' TD1',' TD2',' TD3',' TD4','  Y1','YMAX','ZET1','ZET2','ETX1',00024570
     1    ' TRF',' TRR','  T1',' WN1',' WN2','  R1',' PA0','SN10','SN11'/00024580
C    1    ' TRF',' TRR','  T1',' WN1',' WN2','  R1',' PA0','SN10','SN11',00024580
C    1    'SNSW','DIST','  PL','    ','    ','    ','    ','    ','    ',00024590
      DATA NAMEX
     1  /'SNSW','DIST','  PL','    ','    ','    ','    ','    ','    ',00024590
     1    '    ',' K1F',' K2F',' K3R',' K4R','DLFN','DLRN','TRSW',       00024600
     1    '    ','    ',' ABS','MTQR','SPCO',' SWG','SWRG',' EK1',' EK2',00024610
     1    'BMPL','BMPS','BMPH','  XB','MOC1','MOC2','MOC3','MOC4','MOC5',00024620
     1    'MOC6',' GC0',' GC1','M1C1','M1C2','M1C3','PTC1','PTC2','KAC1',00024630
     1    'KAC2',' BC1',' BC2','FFE1','FFE2','THE1','THE2','CR1C','CR1T',00024640
     1    'CR2C','CR2T','CR3C','CR3T','CR4C','CR4T','  H1','  H2','  NI',00024650
     1    'AKF1','AKF2','AKF3','AKF4',' BR1',' BR2',' BR3',' BR4',       00024660
     1    'SS11','SS21','SS31','SS41','SS12','SS22','SS32','SS42','SS13',00024670
     1    'SS23','SS33','SS43','SS14','SS24','SS34','SS44','BP11','BP21',00024680
     1    'BP31','BP41','BP12','BP22','BP32','BP42','LB11','LB21','LB31',00024690
     1    'LB41','LB12','LB22','LB32','LB42',' TLF',' ZR0','  LF','BMPN',00024700
     1    'TQB0','TQB1','    ',' TSF',' TSR','TANP',' HFC',' HRC','CRRC',00024710
     1    'AXLE','PSR3','PSR4','ROT','RA0','RA1','RA2','RA3','RA4'/
C    1    'AXLE','PSR3','PSR4'/
      DATA RAD/0.1745329E-1/                                             00024730
      REAL*4       IOUT   ,IN     ,ITMP
      INTEGER*2    RTSW   ,RBSW   ,LDTSW ,OPEN   ,OPDN
  960 FORMAT('1  PARAMETER VALUES - MODEL C - ',20A4,/                  00024740
     1    ('0',5(I4,3X,A4,'=',G12.5,',')))                              00024750
C 960 FORMAT('1  PARAMETER VALUES - MODEL C - '20A4,/                   00024740
C    1    ('0',5(I4,3X,A4,'='G12.5,',')))                               00024750
  999 FORMAT('0AND THERE YOU ARE')                                      00024760
      DO 777 I=1,8                                                      00024770
C     CALL DISARM(I-1)                                                  00024780
  777 CONTINUE                                                          00024790
      TQFMAX=-1.E20                                                     00024800
      TQRMAX=-1.E20                                                     00024810
      AF01=PARAM(202)                                                   00024820
      AF02=PARAM(203)                                                   00024830
      AF03=PARAM(204)                                                   00024840
      AF04=PARAM(205)                                                   00024850
      AF05=PARAM(206)                                                   00024860
      AF06=PARAM(207)                                                   00024870
      GAMC0=PARAM(208)                                                  00024880
      GAMC1=PARAM(209)                                                  00024890
      AF11=PARAM(210)                                                   00024900
      AF12=PARAM(211)                                                   00024910
      AF13=PARAM(212)                                                   00024920
      AP1=PARAM(055)                                                    00024930
      AP2=PARAM(213)                                                    00024940
      AP3=PARAM(214)                                                    00024950
      AP4=PARAM(215)                                                    00024960
      AP5=PARAM(216)                                                    00024970
      RTC1=PARAM(217)
      RTVMAX=-1.E20
      RTC2=PARAM(218)
      FFF1=PARAM(219)*.01745329
      FFF2=PARAM(220)*.01745329
      PSIMAX=-1.E20
      THE1=PARAM(221)*.01745329
      THE2=PARAM(222)*.01745329
      PSIM=PSIIN*RAD
      XEND=TEND
      TMAX1=-1.E20
      TMAX2=-1.E20
      TMAX3=-1.E20
      TIME25=0.0
      TIME10=0.0
      D1=-6.0E-6
      D2=0.009
      D3=0.0001
      F4=-0.16
      F5=-0.46
      F6=10.4
      ANUMO=0.0
      ADENO=0.0
      RMAX=-1.E20
      PSI5=0.0
      DSWMAX=-1.E20
      PHIMAX=-1.E20
      ETAMAX=-1.E20
      ITSC=PARAM(88)+0.5
      DEL1DT=DEL1DN
      DEL2DT=DEL2DN
      DEL3DT=DEL3DN
      DEL1=DEL1IN
      DEL2=DEL2IN
      DEL3=DEL3IN
      PHIRD=PHIDN*RAD
      PHIR=PHIRN*RAD
      DELEW1=DEW1IN*RAD
      DELEW2=DEW2IN*RAD
      DDD=0.0
      DD=0.0
      U1P=U1PIN
      U2P=U2PIN
      U3P=U3PIN
      U4P=U4PIN
      S1P=S1PIN
      S2P=S2PIN
      S3P=S3PIN
      S4P=S4PIN
      TTA=0.0
      PESLP=PEL/0.2
      QUAN1=0.0
      QUAN2=0.0
      QUAN3=0.0
      QUAN4=0.0
      F1F1=CFP*SIGN(1.,DEL1DT)
      F1F2=CFP*SIGN(1.,DEL2DT)
      IF(DEL1.LT.DFC)GO TO 1
      IF(DEL1.GT.DF1)GO TO 2
      F2F1=AKF*DEL1
      GO TO 3
    1 F2F1=AKF*(ALAMF*DEL1-(ALAMF-1.)*OFC)
      GO TO 3
    2 F2F1=AKF*(ALAMF*DEL1-(ALAMF-1.)*OFT)
    3 IF(DEL2.LT.OFC)GO TO 4
      IF(DEL2.GT.OFT)GO TO 5
      F2F2=AKF*DEL2
      GO TO 6
    4 F2F2=AKF*(A-AMF*DEL2-(ALAMF-1.0)*OFC)
      GO TO 6
    5 F2F2=AKF*(ALAMF*DEL2-(ALAMF-1.)*OFT)
    6 S1P=-CF*DEL1DT-F1F1-F2F1+(RF*(DEL2-DEL1))/TF**2
      S2P=-CF*DEL2DT-F1F2-F2F2-(RF*(DEL2-DEL1))/TF**2
      IF( PARAM(287).EQ.2. ) GO TO 106
      ZETA3=TS*PHIR/2.+DEL3
      ZETA3D=TS*PHIRD/2.+DEL3DT
      ZETA4=-ZETA3+2.*DEL3
      ZETA4D=-ZETA3D+2.*DEL3DT
      IF(ZETA3.LT.ORC)GO TO 7
      IF(ZETA3.GT.ORT)GO TO 8
      F2R3=AKR*ZETA3
      GO TO 9
  106 ZETA3=DEL3
      ZETA4=DEL4
      ZETA3D=DEL3DT
      ZETA4D=DEL4DT
      GO TO 9
    7 F2R3=AKR*(ALAMR*ZETA3-(ALAMR-1.)*ORC)
      GO TO 9
    8 F2R3=AKR*(ALAMR*ZETA3-(ALAMR-1.)*ORT)
    9 IF(ZETA4.LT.ORC)GO TO 10
      IF(ZETA4.GT.ORT)GO TO 11
      F2R4=AKR*ZETA4
      GO TO 12
   10 F2R4=AKR*(ALAMR*ZETA4-(ALAMR-1.)*ORC)
      GO TO 12
   11 F2R4=AKR*(ALAMR*ZETA4-(ALAMR-1.)*ORT)
   12 F1R3=CRP*SIGN(1.,ZETA3D)
      F1R4=CRP*SIGN(1.,ZETA4D)
      S3P=-CR*ZETA3D-F1R3-F2R3-RR*PHIR/TS
      S4P=-CR*ZETA4D-F1R4-F2R4+RR*PHIR/TS
      P=PIN*RAD
      PO=P
      Q=QIN*RAD
      QO=Q
      R=RIN*RAD
      RO=R
      U=UIN
      UO=U
      V=VIN
      VO=VIN
      W=WIN
      WO=WIN
      X=XIN
      XO=XIN
      Y=YIN
      YO=YIN
      Z=ZIN
      ZO=ZIN
      THE=THEIN*RAD
      THEO=THE
      PHI=PHIIN*RAD
      PHIO=PHI
      PSI=PSIIN*RAD
      PSIO=PSI
C     IF(PPRT.NE.0.0)PRINT 960,(ALPH(I),I=1,20),((K,NAME(K),PARAM(K)),
C    1   K=1,N1)
      IF(PPRT.NE.0.0)WRITE(LPTR,960)(ALPH(I),I=1,20),((K,NAME(K),
     1 PARAM(K)),K=1,N1)
  940 FORMAT(10G12.5)
      TIME=0.0
      DT=0.0
  998 FORMAT('0',8E15.6)
C 998 FORMAT('0',8F15.6)
      THSW=0
      XB(1)=PARAM(201)
      NBMP=PARAM(277)+0.5
      IF(NBMP.LT.2)GO TO 4321
      DO 5432 I=2,NBMP
      XB(I)=XB(I-1)+PARAM(199)
 5432 CONTINUE
 4321 CONTINUE
      CALL SBPG22
C     DO 20 I=1,48
C     CALL STIND(IERR,IDAC(I),IOUT(I))
   20 CONTINUE
      CALL LBDAEP(00,39,IOUT,ILBERR)
      CALL TLDA
      WRITE(LPTR,69)(IDAC(I),IOUT(I),I=1,48)
   69 FORMAT(1H1,' DAC #',5X,'DAC VALUE',///,(I3,5X,F12.5))
C     CALL TSCAL(IERR,ITSC)
      CALL STCO(ITSC,ISTCOE)
C     AD-4 & 680 CODE SAME FOR ITSC=0,1.   2,3 GIVES ERROR  CHECK
C      PARAM 88
      DT=DTIN
C     CALL INMUX(IERR,1)
      ISW1=0
      ISW7=0
C     CALL ARM(2)
C     CALL ARM(4)
C     CALL ARM(5)
C     CALL ARM(7)
      RETURN
      END
```

```
C      SUBROUTINE SBPG7                                                  00004959
       SUBROUTINE SBPG7                                                  00004960
       COMMON/APL/ OPEN  ,RTSW  ,LDTSW ,RBSW
       COMMON/DEVICE/KEYBD,ITTY,ICDRD,LPTR
       COMMON/SP7BLK/N1,N2,IPOT(244),IPOTAD(244),PARAM(300)              00004970
       INTEGER YES/'YES '/,NO/'NO'/
       INTEGER*2   RTSW  ,RBSW  ,LDTSW ,OPEN  ,OPNN
       REAL*4      IOUT  ,IN    ,ITMP
       DIMENSION ICPOT(244),ICPOTA(244),IPOLD(244),MPOT(244),MPOTAD(244) 00004990
       DIMENSION JDATE(3)
       DATA NINES/99999999/                                              00005210
       DATA ISW/0/                                                       00005220
C      LI,7     X'FF00'                                                  00005230
C      WD,7     X'1102'                                                  00005240
       CALL SAMO(6,ISAMOE)                                               00005250
       IDEV = KEYBD                                                      00005280
       IF(OPEN.EQ.1) GO TO 75
       CALL IDATE(JDATE)
   75  CONTINUE
       CALL CLOCK(ITIME)
       IHOURS =ITIME/360000
       IDIF   =ITIME-(IHOURS*360000)
       MINUTE =IDIF/6000
       WRITE(ITTY,6969)(JDATE(I),I=1,3),IHOURS,MINUTE
 6969  FORMAT (' DATE  ',I2,'/',I2,'/',I2,5X,'TIME',1X,I2,':',I2,///)
       IF( OPEN.EQ.1 ) GO TO 21                                          00005281
       WRITE(ITTY,10)                                                    00005290
C      CALL IC(IERR)                                                     00005250
C      CALL LOAD(IERR)                                                   00005260
C      CALL HOFF(IERR,0)                                                 00005270
C      IDEV=101                                                          00005280
C      WRITE (102,10)                                                    00005290
   10  FORMAT (' INITIAL POTSET?')                                       00005300
C      READ (101,20) IN1                                                 00005310
       READ(KEYBD,20)IN1                                                     5310
   20  FORMAT (A3)                                                       00005320
       IF (IN1.EQ.YES) GO TO 2000                                        00005330
   31  CONTINUE
  32   WRITE(ITTY,33)
   33  FORMAT(' END OF CARD DATA')
   34  CONTINUE
   21  WRITE( ITTY,30 )                                                  00005340
C      WRITE (102,30)                                                    00005340
   30  FORMAT (' INPUT FROM CONSOLE?')                                   00005350
C      READ (101,20) IN2                                                 00005360
C      IDEV=101                                                          00005370
       READ(KEYBD,20)IN2                                                    5360
       IF((IN2.NE.YES).AND.(IN2.NE.NO)) GO TO 55
       GO TO 65
   55  WRITE(ITTY,66)
   66  FORMAT (' TYPING ERROR, TRY AGAIN')
       GO TO 31
   65  CONTINUE
       IDEV = KEYBD                                                      00005370
   45  CONTINUE
       WRITE( ITTY,11 )                                                  00005371
   11  FORMAT(' RTM REQUIRED?')                                          00005372
       READ( KEYBD,20 )IOPNSW                                            00005373
       IF(IOPNSW.EQ.YES ) RTSW = 1                                       00005374
```

```
      IF(IOPNSW.EQ.NO ) RTSW=0                                          00005375
      IF((IOPNSW.NE.YES).AND.(IOPNSW.NE.NO)) GO TO 25
      GO TO 35
   25 WRITE(ITTY,66)
      GO TO 45
   35 CONTINUE
C     IF (IN2.NE.YES) IDEV=105                                          00005380
      IF (IN2.NE.YES) IDEV=ICDRD                                        00005380
      WRITE( ITTY,12 )
   12 FORMAT(' NOPARM(I3,1X)  PARVAL(G20.6)')
C1000 READ (IDEV,50) NOPARM,PARVAL                                      00005390
 1000 READ(IDEV,50,END=32) NOPARM,PARVAL
   50 FORMAT (I3,1X,G20.6)                                              00005400
      IF (NOPARM.GT.300) GO TO 1200                                     00005410
      IF (NOPARM.LE.N1) GO TO 1100                                      00005420
C     WRITE (102,60)                                                    00005430
      WRITE(ITTY,60)                                                    00005430
   60 FORMAT (' PARAMETER NUMBER EXCEEDS LIMIT')                        00005440
      GO TO 1000                                                        00005450
 1100 PARAM(NOPARM)=PARVAL                                              00005460
      GO TO 1000                                                        00005470
 1200 IF (NOPARM.LE.307) GO TO 1210                                     00005480
C     WRITE (102,700)                                                   00005490
      WRITE(ITTY,700)                                                   00005490
  700 FORMAT(' PARAMETER NUMBER LARGER THAN 307 NOT ALLOWED--TRY AGAIN')00005500
      GO TO 1000                                                        00005510
 1210 IF (NOPARM.NE.307) GO TO 1250                                     00005520
      CALL READCD                                                       00005530
      GO TO 1900                                                        00005540
 1250 CALL POTSET                                                       00005550
      IF (ISW.EQ.1) GO TO 1500                                          00005560
      ID=0                                                              00005570
      DO 1300 J =1,244                                                  00005580
      IPOLD(J)=IPOT(J)                                                  00005590
      IF (IPOT(J).EQ.NINES) GO TO 1300                                  00005600
      ID=ID+1                                                           00005610
      ICPOT(ID)=IPOT(J)                                                 00005620
      ICPOTA(ID)=IPOTAD(J)                                              00005630
 1300 CONTINUE                                                          00005640
      IF (ID.EQ.N2) GO TO 1350                                          00005650
C     WRITE (102,70) N2                                                 00005660
      WRITE(ITTY,70) N2                                                 00005660
   70 FORMAT (' POTSET REQUIRES ', I3, ' POT VALUES')                   00005670
 1350 ISW=1                                                             00005680
      GO TO 1550                                                        00005690
 1500 ID=0                                                              00005700
      DO 1520 J=1,244                                                   00005710
      IF (IPOT(J).EQ.IPOLD(J)) GO TO 1520                               00005720
      IPOLD(J)=IPOT(J)                                                  00005730
      ID=ID+1                                                           00005740
      ICPOT(ID)=IPOT(J)                                                 00005750
      ICPOTA(ID)=IPOTAD(J)                                              00005760
 1520 CONTINUE                                                          00005770
 1550 ISW1=NOPARM-300                                                   00005780
      GO TO (1800,1600,1700,1700,1700,1780),ISW1                        00005790
C1600 WRITE (108,100)                                                   00005800
 1600 WRITE (LPTR,100)                                                  00005800
  100 FORMAT ('1'/'0'/'0CHANGED POT ADDRESSES AND VALUES:'/'0'/)        00005810
      K=50                                                              00005820
      IF (ID.LT.50) K=ID                                                00005830
      ITEMP=ID/50
      DO 1650 L=1,K
      LLAST=50*ITEMP+L
      IF (LLAST.GT.ID) LLAST=LLAST-50
C     WRITE (108,150) (ICPOTA(J),ICPOT(J),J=L,LLAST,50)
      WRITE(LPTR,150) (ICPOTA(J),ICPOT(J),J=L,LLAST,50)
  150 FORMAT (' ', 4(I4,6X,I6,14X))
 1650 CONTINUE
      GO TO 1800
C1700 WRITE (108,200)
 1700 WRITE(LPTR,200)
  200 FORMAT ('1'/'0'/'0POT ADDRESSES AND VALUES:'/'0'/)
      IG=0
      DO 1720 J=1,244
      IF (IPOT(J).EQ.NINES) GO TO 1720
      IG=IG+1
      MPOT(IG)=IPOT(J)
      MPOTAD(IG)=IPOTAD(J)
 1720 CONTINUE
      K=50
      IF (IG.LT.50) K=IG
      ITEMP=IG/50
      DO 1750 L=1,K
      LLAST=50*ITEMP+L
      IF (LLAST.GT.IG) LLAST=LLAST-50
C     WRITE (108,150) (MPOTAD(J),MPOT(J),J=L,LLAST,50)
      WRITE(LPTR,150) (MPOTAD(J),MPOT(J),J=L,LLAST,50)
 1750 CONTINUE
      IF (ISW1.EQ.3) GO TO 1800
C     WRITE (108,300)(J,PARAM(J),J=1,N1)
      WRITE(LPTR,300)(J,PARAM(J),J=1,N1)
  300 FORMAT ('1'/'0PARAMETER NO.', 5X,'PARAMETER VALUE'/'0'/
     1(' ', 6X, I3, 11X, G13.5))
      IF (ISW1.EQ.4) GO TO 1800
      IF (ISW1.EQ.5) GO TO 1900
C1780 WRITE (106,400) (J,PARAM(J),J=1,N1)
 1780 WRITE(ICDPN,400)(J,PARAM(J),J=1,N1)
  400 FORMAT (I3,1X,G20.6)
      GO TO 1900
 1800 CONTINUE
   87 FORMAT (6G20.6)
C     CALL INITA(IERR,0)
C     IF (IERR.NE.0) GO TO 1800
      IF (ID.EQ.0) GO TO 1900
      DO 1850 J=1,ID
C     CALL STIND(IERR,ICPOTA(J),ICPOT(J))
      CALL SPOT(ICPOTA(J),ICPOT(J),03,IERR)
 1850 CONTINUE
C1900 CALL INITA(IERR,0)
C     IF (IERR.NE.0) GO TO 1900
 1900 CONTINUE
 1950 CONTINUE
      CALL SAMO(6,ISAMOF)
C     LI,7    X'FF00'
C     WD,7    X'1202'
      RETURN
 2000 CONTINUE
         OPEN   = 1
      IF ((N1.GT.300) .OR. (N2.GT.196)) GO TO 9000
C     WRITE (102,500) N1,N2
```

```
      WRITE (ITTY,500) N1,N2                                            00006340
  500 FORMAT (' ', I3, ' PARAMETERS AND ', I3, ' POTS REQUIRED'/        00006350
     1' INPUT FROM CARDS?')                                             00006360
C     READ (101,20),IN2                                                 00006370
      READ(KEYRD,20)IN2                                                   6370
C     IDEV=105                                                          00006380
C     IF (IN2.NE.YES) IDEV=101                                          00006390
      IDEV = ICDRD                                                      00006380
      IF (IN2.NE.YES) IDEV=KEYRD                                        00006390
C9000 WRITE (108,600)                                                    00006450
 9000 WRITE(LPTP,600)                                                   00006450
      ISW=0                                                             00006400
      DO 3000 J=1,244                                                   00006410
      IPOT(J)=NINES                                                     00006420
 3000 CONTINUE                                                          00006430
      GO TO 1000                                                        00006440
  600 FORMAT (' ONLY 300 PARAMETERS AND 244 POTS ARE ALLOWED')          00006460
C     STOP                                                              00006470
      RETURN                                                            00006470
      END                                                               00006480
```

```
C       SUBROUTINE RTMON
        SUBROUTINE RTMON
C
        COMMON/APL/ OPEN   ,RTSW   ,LDTSW ,RBSW
        COMMON /RBBLK/ AD2RB ,AD1RB ,CLSRB ,CLRRB ,ICRB   ,OPRB ,PILRB
        COMMON /RBBLK/ TCNBUF,TIMBUF,LDARB ,TDARB ,PILRB1
        COMMON /ECBBLK/PILECB,TCNECB,TIMECB,ADAECB,TDAECB
        COMMON /ECBBLK/AD2ECB,AD1ECB,CLSECB,CLRECB,ICECB ,OPECB
        COMMON /ECBBLK/OSECB ,DONECB
        COMMON/DEVICE/KEYBD,ITTY,ICDRD,LPTR
        COMMON/HHHH/H1,H2,H3,H4
        COMMON/SPLTAX/SPSR3,CPSR3,SPSR4,CPSR4,SCR3,SCR4,TBCR3,TBCR4,TBSR3,
       1   TBSR4,TRSR3,TRSR4,TRCR3,TRCR4,PSR3,PSR4,IAX
        COMMON/TABLES/PHIFNT(7),THEFNT(7),PSIFNT(7),PHIRR(7),THERR(7),
       1   PSIRR(7)
        COMMON/EXTRA/              UOUT,ROUT,        PSIOUT,EXTAB(100),
       1   DLYTB(100),RTAB(100)
        COMMON/FRIDAY/BTVMAX,BTVTB(100)
        COMMON/THINGS/TMAX1,TMAX2,TMAX3,TQRMAX,TQFMAX,PSIMAX,ONER
        COMMON/FEES/FEE1,FEE2,THE1,THE2
        COMMON/DELS/DELSWC
        COMMON/CLEAN/ONEOA,ONEOD,PBSW,PB
        COMMON/NUTS/TOO
        COMMON/XYZ/NUMBR
        COMMON/QUANTS/QUAN1,QUAN2,QUAN3,QUAN4
        COMMON/EFFS/ANUM,ADEN,ANUMDT,ADENDT,ANUMO,ADENO,ANUMDO,ADENDO,
       1   ANOUT,ADOUT
        COMMON/XBS/XB(12),NS(4,12),DELX(4),XI(4),NNN
        COMMON/NEWER/TIME25,TIME10,RMAX,ETAMAX,PSI5,PHIMAX,DSWMAX
        COMMON/VARS/P,Q,R,U,V,W,X,Y,Z,THE,PHI,PSI,PO,QO,RO,UO,VO,WO,XO,
       1    YO,ZO,THEO,PHIO,PSIO
        COMMON/UVW/VC,UIN
        COMMON/EES/E1,E2,E3,E4,E5,E6
        COMMON/ZILCH/TQMAXP,AKTQP
        COMMON/INOUT/INA(32),IOUTA(48),IN(32),IOUT(48),ISW1,ISW7,SFIN(32),
       1     SFOUT(48),IPRT,ITMP(48)
        COMMON/ANALOG/DEL1DT,DEL2DT,DEL3DT,DEL1,DEL2,DEL3,PHIRD,PHIR,
       1    DELFW1,DELFW2,U1P,U2P,U3P,U4P,S1P,S2P,S3P,S4P,TTA,PFSLP
        COMMON/COMBLK/AIXP,SM,AIYP,AIXZP,GAM1,GAM2,GAM3,AIXBR,AIYBR,
       1    AIZBR,A1,A2,AIXZBR,A12,E1,E2,E3,DELTA,GV1,GV2,GP1,GP2,GR1,
       1    GR2,CIP,CIVP,RZF,RZR,A2T,CA20,CA23,TIME,DT,      ANGNL,ANGNLO
       1    ,TR02,TF02,TS02,G,THRD,TWN7
        COMMON/DDD/DIN,DDTO,DDT,TIME5,DDO,DD,DOUT
        COMMON/OPSW/IHSW
        COMMON/SP7BLK/N1,N2,IPOT(244),IPOTAD(244),PARAM(300)
        COMMON/MICKEY/IRUNTB(100),VTB(100),SNLTB(100),SNRTB(100),DSWTB(100
       1 ),BTTB(100),DTB(100),EFFTB(100),DYTB(100),PSITB (100),NRUN,
       1   YSPEC,PSIM,XPF
       1   ,TMPTB(100)
        COMMON/NONAME/XEND,O,EXIT2
        DIMENSION CSI(4),XBM(4),SLP(4)
C
        REAL*8       BUFF(8)     ,PILRB(3)    ,LDARB(21)   ,TCNBUF(8)
        REAL*8       TIMBUF(8)
        REAL*8       SAVE2(16)   ,PILRB1(3)   ,AD2RB(6)    ,AD1RB(6)
        REAL*8       CLRRB(6)    ,OPRB(6)     ,TDARB(6)
        REAL*8       SAVE0(16)   ,SAVE1(16)   ,CLSRB(6)    ,ICRB(6)
C
        REAL*4       IOUT   ,IN     ,ITMP
```

```
      REAL*4      ADC2(04)   ,ADC1(24)
C
      INTEGER*4   TIMCAN     ,TCNECB     ,TIMECB
      INTEGER*4   CONSL/01/  ,PILECB     ,ADAECB     ,TDAECB
      INTEGER*4   RCL0/00/   ,CLRECB     ,IMODOP/04/ ,OPECB
      INTEGER*4               EVTRET/02/ ,TIMINT/120000/
      INTEGER*4   F2/20/     ,L2/23/     ,FIRST/00/  ,LAST/39/
      INTEGER*4               NONE/00/   ,AD2ECB     ,AD1ECB
      INTEGER*4   SCL0/00/   ,CLSECB     ,IMODIC/06/ ,ICECB
      INTEGER*4     MONE/-1/   ,F1/00/     ,L1/23/
      INTEGER*4   TDAECB     ,STATUS
      INTEGER*4   OSECB      ,DONECB
C
      INTEGER*2   PILIST(2)/1,0/,EVTLST/1/
      INTEGER*2               NUMEVT/03/ ,ZERO/00/
      INTEGER*2   UNIT/19/
      INTEGER*2   TWO  /02/ ,ONE/01/
      INTEGER*2   RTSW  ,RBSW  ,LDTSW ,OPEN  ,OPDN
C
      EQUIVALENCE (ADC1(24),IN(24))      ,(ADC2(1),IN(25))
C
      EXTERNAL        INIT,CART ,ENDRUN,HYBINT
C
      IF( RBSW.EQ.1 ) GO TO 200
         CALL BLJCB( 'J007',OSECB,NUMEVT,NONE )
         CALL DEFFP( INIT,SAVE0,ZERO,'NONE','NO' )
         CALL DEFFP( ENDRUN,SAVE1,ZERO,'NONE','NO' )
         CALL DEFFP( CART,SAVE2,ZERO,'NONE','NO' )
         CALL CRBCRB( F1,L1,ADC1,AD1RB,AD1ECB,CONSL )
         CALL CRBCRB( F2,L2,ADC2,AD2RB,AD2ECB,CONSL )
         CALL TLDARB( TDARB,TDAECB,CONSL )
         CALL SSCLRB( SCL0,CLSRB,CLSECB,CONSL )
         CALL RSCLRB( RCL0,CLRRB,CLRECB,CONSL )
         CALL SAMORB( IMODIC,ICRB,ICECB,CONSL )
         CALL SAMORB( IMODOP,OPRB,OPECB,CONSL )
         OPDN   = 0
         RBSW   = 1
  200 CONTINUE
      IF( RTSW.EQ.0 ) GO TO 210
      IF( OPDN.EQ.1 ) GO TO 205
         OPDN   = 1
         CALL RTOPN
         CALL RTACT( ZERO,'J007' )
  205 CONTINUE
         CALL DEFPR( UNIT,HYBINT,'J007' )
         LDTSW  = 0
         OSECB  = 0
         CALL RTACT( TWO ,'J007' )
         CALL WAITRT( OSECB )
         CALL WAITBU( 200 )
         CALL DEFPR( UNIT,MONE,'J007' )
      GO TO 215
  210 CONTINUE
         CALL LRDAFP( FIRST,LAST,IOUT,IERR )
         CALL TLDA
         CALL RSCL( SCL0,ICLERR )
         CALL CRBCFP( F2,L2,ADC2,ICRBCE )
         CALL SSCL( RCL0,ICLERR )
         CALL CRBCFP( F1,L1,ADC1,ICRBCE )
         CALL SRPG2
```

```
215 CONTINUE
    RETURN
    END
```

```
C       SUBROUTINE RTIME
        SUBROUTINE RTIME
C
        COMMON/APL/ OPEN   ,RTSW   ,LDTSW ,RBSW
        COMMON /RRBLK/ AD2RB ,AD1RB ,CLSRB ,CLRRB ,ICRB   ,OPRB ,PILRB
        COMMON /RRBLK/ TCNBUF,TIMBUF,LDARB ,TDARB ,PILRB1
        COMMON /FCBBLK/PILFCB,TCNFCB,TIMECB,ADAFCB,TDAFCB
        COMMON /FCBBLK/AD2FCB,AD1FCB,CLSFCB,CLRFCB,ICFCB ,OPFCB
        COMMON /FCBBLK/OSFCB ,DONECB
        COMMON/DEVICE/KEYBD,ITTY,ICDRD,LPTR
        COMMON/HHHH/H1,H2,H3,H4
        COMMON/SPLTAX/SPSR3,CPSR3,SPSR4,CPSR4,SCR3,SCR4,TBCR3,TBCR4,TBSR3,
     1    TBSR4,TRSR3,TRSR4,TRCR3,TRCR4,PSR3,PSR4,IAX
        COMMON/TABLES/PHIENT(7),THEENT(7),PSIENT(7),PHIRR(7),THERR(7),
     1    PSIRR(7)
        COMMON/EXTRA/               UOUT,ROUT,          PSIOUT,EXTAB(100),
     1    DLYTB(100),RTAB(100)
        COMMON/FRIDAY/BTVMAX,BTVTB(100)
        COMMON/THINGS/TMAX1,TMAX2,TMAX3,TQRMAX,TQEMAX,PSIMAX,ONER
        COMMON/FEES/FEE1,FEE2,THE1,THE2
        COMMON/DELS/DELSWC
        COMMON/CLEAN/ONEOA,ONEOD,PRSW,PR
        COMMON/NUTS/TOO
        COMMON/XYZ/NUMBR
        COMMON/QUANTS/QUAN1,QUAN2,QUAN3,QUAN4
        COMMON/EEES/ANUM,ADEN,ANUMDT,ADENDT,ANUMO,ADENO,ANUMDO,ADENDO,
     1    ANOUT,ADOUT
        COMMON/XBS/XB(12),NS(4,12),DELX(4),XI(4),NNN
        COMMON/NEWER/TIME25,TIME10,RMAX,ETAMAX,PSI5,PHIMAX,DSWMAX
        COMMON/VARS/P,Q,R,U,V,W,X,Y,Z,THE,PHI,PSI,PO,QO,RO,UO,VO,WO,XO,
     1     YO,ZO,THEO,PHIO,PSIO
        COMMON/UVW/VC,UIN
        COMMON/EES/O1,O2,O3,E4,E5,E6
        COMMON/ZILCH/TQMAXP,AKTQP
        COMMON/INOUT/INA(32),IOUTA(48),IN(32),IOUT(48),ISW1,ISW7,SFIN(32),
     1      SFOUT(48),IPRT,ITMP(48)
        COMMON/ANALOG/DEL1DT,DEL2DT,DEL3DT,DEL1,DEL2,DEL3,PHIRD,PHIR,
     1     DELFW1,DELFW2,U1P,U2P,U3P,U4P,S1P,S2P,S3P,S4P,TTA,PFSLP
        COMMON/COMBLK/AIXP,SM,AIYP,AIXZP,GAM1,GAM2,GAM3,AIXBR,AIYBR,
     1     AIZBR,A1,A2,AIXZBR,A12,E1,E2,E3,DELTA,GV1,GV2,GP1,GP2,GR1,
     1     GR2,CIP,CIVP,RZF,RZR,A2T,CA20,CA23,TIME,DT,      ANGNL,ANGNLO
     1     ,TRO2,TFO2,TSO2,G,THRD,TWN7
        COMMON/DDD/DIN,DDTO,DDT,TIME5,DDO,DD,DOUT
        COMMON/OPSW/IHSW
        COMMON/SP7BLK/N1,N2,IPOT(244),IPOTAD(244),PARAM(300)
        COMMON/MICKEY/IRUNTB(100),VTB(100),SNLTB(100),SNRTB(100),DSWTB(100
     1  ),RTTB(100),DTB(100),EEETB(100),DYTB(100),PSITB (100),NRUN,
     1    YSPFC,PSIM,XPF
     1    ,TMPTB(100)
        COMMON/NONAME/XEND,Q,EXIT2
C
        DIMENSION CSI(4),XBM(4),SLP(4)
C
        REAL*8        BUFF(8)     ,PILRB(3)    ,LDARB(21)    ,TCNBUF(8)
        REAL*8        TIMBUF(8)
        REAL*8        SAVE0(16)   ,SAVE1(16)   ,CLSRB(6)     ,ICRB(6)
        REAL*8        SAVE2(16)   ,PILRB1(3)   ,AD2RB(6)     ,AD1RB(6)
        REAL*8        CLRRB(6)    ,OPRB(6)     ,TDARB(6)
        REAL*8       BUFF1(8)
```

```
C
      REAL*4      IOUT  ,IN      ,ITMP
      REAL*4      ADC2(04)  ,ADC1(24)
C
      INTEGER*4   TIMCAN    ,TCNECB     ,TIMECB
      INTEGER*4   CONSL/01/ ,PILECB     ,ADAECB     ,TDAECB
      INTEGER*4              EVTRET/02/ ,TIMINT/120000/
      INTEGER*4   RCL0/00/  ,CLRECB     ,IMODOP/04/ ,OPECB
      INTEGER*4   F2/20/    ,L2/23/     ,FIRST/00/  ,LAST/39/
      INTEGER*4              NONE/00/   ,AD2ECB     ,AD1ECB
      INTEGER*4   SCL0/00/  ,CLSECB     ,IMODIC/06/ ,ICECB
      INTEGER*4     MONE/-1/  ,F1/00/     ,L1/23/
      INTEGER*4   TDAECB    ,STATUS
      INTEGER*4   OSECB     ,DONECB
      INTEGER*4  PILCB1
C
      INTEGER*2   PILIST(2)/1,0/,EVTLST/1/
      INTEGER*2              NUMEVT/03/ ,ZERO/00/
      INTEGER*2   UNIT/19/
      INTEGER*2   TWO  /02/ ,ONE/01/
      INTEGER*2   RTSW  ,RBSW  ,LDTSW ,OPEN  ,OPDN
C
      EQUIVALENCE (ADC1(24),IN(24))     ,(ADC2(1),IN(25))
C
C     EVENT 0
C
      ENTRY INIT
         CALL PGET( PILIST, 0   ,'J007',PILRB1 )
         PILECB = 0
         CALL PCAN( PILIST,BUFF,PILECB )
         CALL HIOCHK( PILECB )
         CALL PEVT( PILIST,EVTLST,'J007',PILRB )
         DONECB = 0
         CALL HWAIT( DONECB )
         CALL PREL( PILIST,'J007',PILRB1 )
         CALL HDONE( 'DN' )
      CALL HEXIT
C
C     EVENT 1
C
      ENTRY ENDRUN
      TCNECB = 0
         CALL RDTIMR( TIMCAN,TCNECB,'CANC',TCNBUF )
         CALL HIOCHK( TCNECB )
         PILECB = 0
         CALL PDAC( PILIST,BUFF,PILECB )
         CALL HIOCHK( PILECB )
         PILECB = 0
         CALL PCAN( PILIST,BUFF,PILECB )
         CALL HIOCHK( PILECB )
      ICECB  = 0
         CALL HIOREQ( ICRB )
         CALL HIOCHK( ICECB )
         CALL RTCAN( TWO   ,STATUS )
         CALL HOSPST( 'EN' )
      CALL HEXIT
C
C     EVENT 2
C
      ENTRY CART
      IF( LDTSW.EQ.1 ) GO TO 230
      PILCB1 = 0
         CALL PACT( PILIST,BUFF1,PILCB1 )
         CALL HIOCHK ( PILCB1 )
      TIMECB = 0
      TIMINT = 1.E07*DT
         CALL LDTIMR( TIMINT,TIMECB,EVTRET,TIMBUF )
      OPECB  = 0
         CALL HIOREQ( OPRB )
         CALL HIOCHK( OPECB )
         LDTSW  = 1
  230 CONTINUE
      CLRECB = 0
         CALL HIOREQ( CLRRB )
         CALL HIOCHK( CLRECB )
      ADAECB = 0
         CALL LADART( FIRST,LAST,IOUT,LDARB ,ADAECB,CONSL )
         CALL HIOCHK( ADAECB )
      TDAECB = 0
         CALL HIOREQ( TDARB )
         CALL HIOCHK( TDAECB )
      AD2ECB = 0
         CALL HIOREQ( AD2RB )
         CALL HIOCHK( AD2ECB )
      CLSECB = 0
         CALL HIOREQ( CLSRB )
         CALL HIOCHK( CLSECB )
      AD1ECB = 0
         CALL HIOREQ( AD1RB )
         CALL HIOCHK( AD1ECB )
         CALL SAPG2
C     APL WILL TERMINATE REAL-TIME RUN IF EITHER
C     CONDITION SHOWN BELOW IS SATISFIED
      O=SQRT(U*U+V*V)                                                   00022694
      IF(( ISW1.EQ.1 ).OR.( ISW7.EQ.1 )) CALL RTACT( ONE    ,'J007' )
      IF( TIME.LE.XEND.AND.O.GE.EXIT2 ) GO TO 250
      CALL RTACT( ONE,'J007' )
  250 CONTINUE
      CALL HEXIT
      RETURN
      END
```

```
C     FUNCTION POLY(DL,TBL)                                             00006489
      FUNCTION POLY(DL,TBL)                                             00006490
      DIMENSION TBL(7)                                                  00006500
      TMP=TBL(7)                                                        00006510
      DO 10 I=1,6                                                       00006520
      TMP=TMP*DL+TBL(7-I)                                               00006530
   10 CONTINUE                                                          00006540
      POLY=TMP*.0174533                                                 00006550
      RETURN                                                            00006560
      END                                                               00006570
C     SUBROUTINE SBPG1                                                  00026459
      SUBROUTINE SBPG1                                                  00026460
      COMMON/DEVICE/KEYBD,ITTY,ICDRD,LPTR
      COMMON/INOUT/INA(32),IOUTA(48),IN(32),IOUT(48),ISW1,ISW7,SFIN(32),00026470
     1      SFOUT(48),IPRT,ITMP(48)                                     00026480
C     LI,7  X'FF00'                                                     00026490
C     WD,7  X'1102'                                                     00026500
C     WRITE (102,1)                                                     00026510
      WRITE(ITTY,1)                                                     00026510
    1 FORMAT('1 INTERRUPT DURING DIGITAL COMPUTE CYCLE')                00026520
      ISW1=1                                                            00026530
      RETURN                                                            00026540
      END                                                               00026550
C     SUBROUTINE SBPG5                                                  00026559
      SUBROUTINE SBPG5                                                  00026560
      COMMON/XYZ/NUMBR                                                  00026570
C     DATA NUMBR/0/                                                     00026580
      NUMBR=0                                                              26580
C     LI,7  X'FF00'                                                     00026590
C     WD,7  X'1102'                                                     00026600
      ISW1=1                                                            00026610
C     CALL IC(IERR)                                                     00026620
      CALL SAMO(6,ISAMOE)                                               00026620
C     CALL LOAD(IERR)                                                   00026630
C     CALL HOFF(IERR)                                                   00026640
      DO 50 I=1,NUMBR                                                   00026650
   50 CONTINUE                                                          00026660
  910 FORMAT('0',I5,8E15.6,/(6X,8E15.6))                                   26670
C     WRITE (102,10) NUMBR                                              00026680
      WRITE(ITTY,10) NUMBR                                              00026680
   10 FORMAT (I16)                                                      00026690
C        LI,7        X'FF00'                                            00026700
C        WD,7        X'1202'                                            00026710
      RETURN                                                            00026720
      END                                                               00026730
C     FUNCTION FF(P)                                                    00010989
      FUNCTION FF(P)                                                    00010990
      COMMON/NEWTBS/TOBF(50),PBF(50),TOBR(50),PBR(50),AMA(50),AFRI(50), 00011000
     1   AMB(50),GAMM(50),AFA(50),GAMF(50),AFB(50),SLPP(50),NTF,NTR,    00011010
     1   NMA,NMB,NFA,NFB                                                00011020
      FF=XINT(P,PBF,TOBF,NTF)                                           00011030
      RETURN                                                            00011040
      END                                                               00011050
C     FUNCTION FR(P)                                                    00011059
      FUNCTION FR(P)                                                    00011060
      COMMON/NEWTBS/TOBF(50),PBF(50),TOBR(50),PBR(50),AMA(50),AFRI(50), 00011070
     1   AMB(50),GAMM(50),AFA(50),GAMF(50),AFB(50),SLPP(50),NTF,NTR,    00011080
     1   NMA,NMB,NFA,NFB                                                00011090
      FR=XINT(P,PBR,TOBR,NTR)                                           00011100
      RETURN
      END
C     FUNCTION FAMII(FRI,GAMI)
      FUNCTION FAMII(FRI,GAMI)
      COMMON/NEWTRS/TQBF(50),PBF(50),TQBR(50),PBR(50),AMA(50),AFRI(50),
     1   AMB(50),GAMM(50),AFA(50),GAMF(50),AFB(50),SLPP(50),NTF,NTR,
     1   NMA,NMB,NFA,NFB
C     TMP1=XINT(FRI,AFRI,AMA,NMA)
C     TMP2=XINT(GAMI,GAMM,AMB,NMB)
C     FAMII=TMP1*TMP2
      TTTT=8.5-0.259*ABS(GAMI)
      IF( TTTT.LE.0.85 ) TTTT=0.85
      FAMII=TTTT
C     FAMII=8.5-0.259*ABS(GAMI)
      RETURN
      END
C     FUNCTION FCSI(GAMI,SLPI)
      FUNCTION FCSI(GAMI,SLPI)
      COMMON/NEWTRS/TQBF(50),PBF(50),TQBR(50),PBR(50),AMA(50),AFRI(50),
     1   AMB(50),GAMM(50),AFA(50),GAMF(50),AFB(50),SLPP(50),NTF,NTR,
     1   NMA,NMB,NFA,NFB
C     TMP1=XINT(GAMI,GAMF,AFA,NFA)
C     TMP2=XINT(SLPI,SLPP,AFB,NFB)
C     FCSI=TMP1*TMP2
      TMP=ABS(SLPI)
      IF(TMP.LE.0.02)GO TO 10
      IF(TMP.LE.0.05)GO TO 12
      IF(TMP.LE.0.10)GO TO 14
      IF(TMP.LE.0.15)GO TO 16
      IF(TMP.LE.0.20)GO TO 18
      IF(TMP.LE.0.30)GO TO 20
      IF(TMP.LE.0.50)GO TO 22
      GO TO 24
   10 FCSI=1.18
      GO TO 51
   12 FCSI=1.18+(0.02-TMP)/3.
      GO TO 51
   14 FCSI=1.24-1.4*TMP
      GO TO 51
   16 FCSI=1.14-0.6*TMP
      GO TO 51
   18 FCSI=1.21-0.8*TMP
      GO TO 51
   20 FCSI=1.27-TMP
      GO TO 51
   22 FCSI=1.29-1.04*TMP
      GO TO 51
   24 FCSI=0.25
   51 RETURN
      END
C     FUNCTION FU0I(GAMI,CVI)
      FUNCTION FU0I(GAMI,CVI)
      COMMON/AFU0/AF01,AF02,AF03,AF04,AF05,AF06,GAMC0
      TEMP=ABS(GAMI)
C     FU0I=1.12-0.27*TEMP
      TTTT=1.12-0.01685*TEMP
      IF(TTTT.LT.0.85)TTTT=0.85
      FU0I=TTTT
      RETURN
      IF(TEMP.LT.GAMC0)GO TO 10
      FUOI=AF03+AF04*CVI+AF05*TEMP+AF06*TEMP*CVI
      RETURN
   10 FUOI=AF01-AF02*CVI
      RETURN
      END
C     FUNCTION GETDEL(X,I,R5,NBMP)
      FUNCTION GETDEL(X,I,R5,NBMP)
      COMMON/XBS/XB(12),NS(4,12),DELX(4),XI(4),NNN
      COMMON/XYZ/NUMBR
      DIMENSION X(4)
      GETDEL=0.0
      IF(NBMP.EQ.0)RETURN
      DO 10 K=1,NBMP
      L=NBMP-K+1
      IF(X(I).LE.XB(L))NS(I,L)=NUMBR+NNN
      IF(X(I).GE.XB(L).AND.NUMBR.LE.NS(I,L))GO TO 20
   10 CONTINUE
      RETURN
   20 GETDEL=R5
      RETURN
      END
C     FUNCTION FU1I(GAMI)
      FUNCTION FU1I(GAMI)
      COMMON/AFU1/AF11,AF12,AF13,GAMC1
      TEMP=ABS(GAMI)
      FU1I=AF13
      IF(TEMP.LT.GAMC1)FU1I=AF11+AF12*TEMP
      FU1I=0.85
      RETURN
      END
C     SUBROUTINE PTBAK(BET,FRI,AKKI,PTBI)
      SUBROUTINE PTBAK(BET,FRI,AKKI,PTBI)
      COMMON/PTBK/AP1,AP2,AP3,AP4,AP5,BTC1,BTC2
      AP5=600.
      AKKI=AP4+FRI/AP5
      TEMP=ABS(BET*57.29578)
      PTBI=AP1
      IF(TEMP.LE.BTC1)RETURN
      PTBI=AP3
      IF(TEMP.GT.BTC2)RETURN
      PTBI=AP1*(1.0-(TEMP-BTC1)*AP2)
      RETURN
      END
C     FUNCTION FNCT(A1,A2,A3,FRI,CVI)
      FUNCTION FNCT(A1,A2,A3,FRI,CVI)
C     FNCT=A1*FRI+A2*CVI+A3
      FNCT=0.85+.0055*CVI
      RETURN
      END
C     FUNCTION XINT(ARG,ARGTB,FUN,NP)
      FUNCTION XINT(ARG,ARGTB,FUN,NP)
      DIMENSION ARGTB(NP),FUN(NP)
      DO 10 I=1,NP
      IF(ARG-ARGTB(I))30,20,10
   10 CONTINUE
      I=NP
   30 IF(I.EQ.1)I=2
      TEMP=(ARG-ARGTB(I-1))/(ARGTB(I)-ARGTB(I-1))
      XINT=FUN(I-1)+(FUN(I)-FUN(I-1))*TEMP
      RETURN
   20 XINT=FUN(I)                                                       00012180
      RETURN                                                            00012190
      END                                                               00012200
C     FUNCTION XYSIN(X)                                                 00022739
      FUNCTION XYSIN(X)                                                 00022740
      X2=X*X                                                            00022750
      XYSIN=X*(1.0-(0.1666667-0.8333333E-2*X2)*X2)                      00022760
      RETURN                                                            00022770
      END                                                               00022780
C     FUNCTION XYCOS(X)                                                 00022789
      FUNCTION XYCOS(X)                                                 00022790
      X2=X*X                                                            00022800
      XYCOS=1.0-(0.5-.4166667E-1*X2)*X2                                 00022810
      RETURN                                                            00022820
      END                                                               00022830
C     FUNCTION AMIN(X,Y)                                                00022839
      FUNCTION AMIN(X,Y)                                                00022840
      IF(X-Y)1,1,2                                                      00022850
    1 AMIN=X                                                            00022860
      RETURN                                                            00022870
    2 AMIN=Y                                                            00022880
      RETURN                                                            00022890
      END                                                               00022900
C     SUBROUTINE READCD                                                 00022909
      SUBROUTINE READCD                                                 00022910
      DIMENSION IADDR(244),IVAL(244)                                    00022920
      COMMON/DEVICE/KEYBD,ITTY,ICDRD,LPTR
      INTEGER YES/'YES '/                                               00022930
      DATA IADDR/3000,3001,3002,3003,3004,3005,3006,3007,3010,3011,3012,00022940
     13013,3014,3015,3016,3017,3020,3021,3022,3023,3024,3025,3026,3027, 00022950
     13030,3031,3032,3033,3034,3035,3036,3037,3040,3041,3042,3043,      00022960
     13044,3045,3046,3047,3050,3051,3052,3053,3054,3055,3056,3057,      00022970
     13064,3065,3066,3067,3070,3071,3072,3073,3074,3075,3076,3077,      00022980
     13100,3101,3102,3103,3104,3105,3106,3107,3110,3111,3112,3113,      00022990
     13114,3115,3116,3117,3120,3121,3122,3123,3124,3125,3126,3127,3130, 00023000
     13131,3132,3133,3134,3135,3136,3137,3140,3141,3142,3143,3144,3145, 00023010
     13146,3147,3150,3151,3152,3153,3154,3155,3156,3157,3160,3161,3162, 00023020
     13163,3164,3165,3166,3167,3170,3171,3172,3173,3174,3175,3176,3177, 00023030
     13240,3241,3242,3243,3244,3245,3246,3247,3250,3251,3252,3253,3254, 00023040
     13255,3256,3257,3264,3265,3266,3267,3270,3271,3272,3273,3274,3275, 00023050
     13276,3277,3320,3321,3322,3323,3324,3325,3326,3327,3330,3331,3332, 00023060
     13333,3334,3335,3336,3337,3340,3341,3342,3343,3344,3345,3346,3347, 00023070
     13350,3351,3352,3353,3354,3355,3356,3357,3364,3365,3366,3367,3370, 00023080
     13371,3372,3373,3374,3375,3376,3377,2220,2221,2222,2223,2224,2225, 00023090
     12226,2227,2230,2231,2232,2233,2234,2235,2236,2237,2200,2201,2202, 00023100
     12203,2204,2205,2206,2207,2210,2211,2212,2213,2214,2215,2216,2217, 00023110
     12300,2301,2302,2303,2304,2305,2306,2307,2310,2311,2312,2313,2314, 00023120
     12315,2316,2317/                                                   00023130
C     CALL READA(IERR,IADDR,IVAL,1,1,208)                               00023140
      PRINT 200, (((IADDR(I+J-1),IVAL(I+J-1)),J=1,208,52),I=1,52)       00023150
  200 FORMAT('1',2X,'COEFFICIENT     VALUE',10X,'COEFFICIENT      VALUE' 00023160
     1,10X,'COEFFICIENT     VALUE',10X, 'COEFFICIENT     VALUE'/' ',4X, 00023170
     1'DEVICE', 25X, 'DEVICE', 25X,'DEVICE', 25X,'DEVICE'/('0',6X,I4,7X,00023180
     1I6,3(14X,I4,7X,I6)))                                              00023190
C     WRITE (102,300)                                                   00023200
      WRITE(ITTY,300)                                                   00023200
  300 FORMAT ('DO YOU WANT POT SETTINGS AND ADDRESSES PUNCHED?  IF SO,  00023210
     1TYPE ''YES'' ,THEN N/L.')                                         00023220
      READ(KEYBD,400) IN1                                               00023230
C     READ (101,400) IN1                                                00023230
```

```
  400 FORMAT (A3)                                                      00023240
      IF (IN1.NE.YES) GO TO 700                                        00023250
      DO 600 I=1,208                                                   00023260
C     WRITE (106,500) IADDR(I),IVAL(I)                                 00023270
      WRITE(ICDPN,500)IADDR(I),IVAL(I)                                 00023270
  500 FORMAT (I4,6X,I5)                                                00023280
  600 CONTINUE                                                         00023290
  700 RETURN                                                           00023300
      END                                                              00023310
      FUNCTION TAN(X)                                                  00023320
      TAN=SIN(X)/COS(X)                                                00023330
      RETURN                                                           00023340
      END                                                              00023350
```

3. PRESENTED HERE ARE THE ANALOG COMPUTER DIAGRAMS

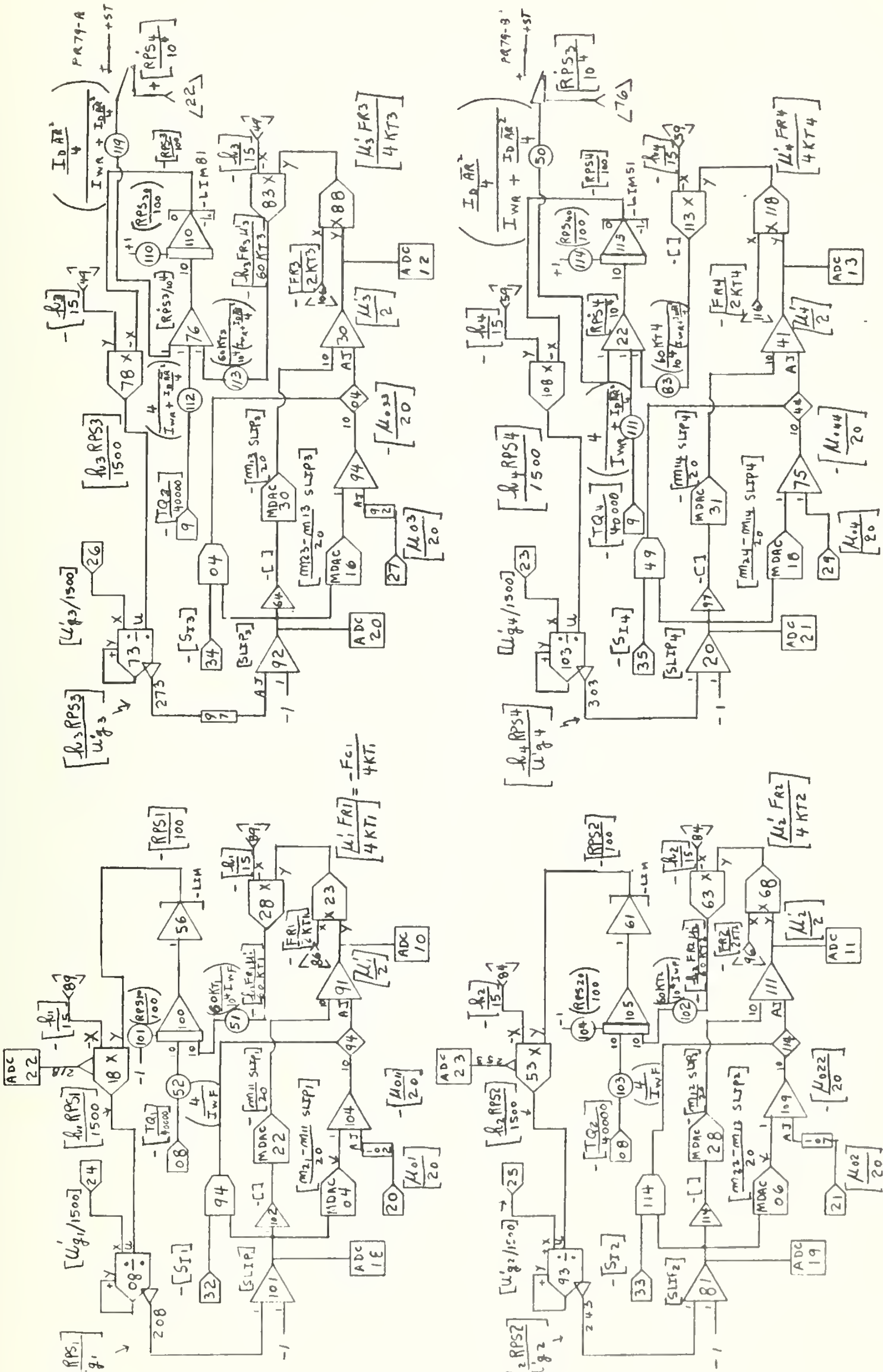


Fig. B-1 ANALOG COMPUTER DIAGRAM – ROTATIONAL WHEEL DYNAMICS

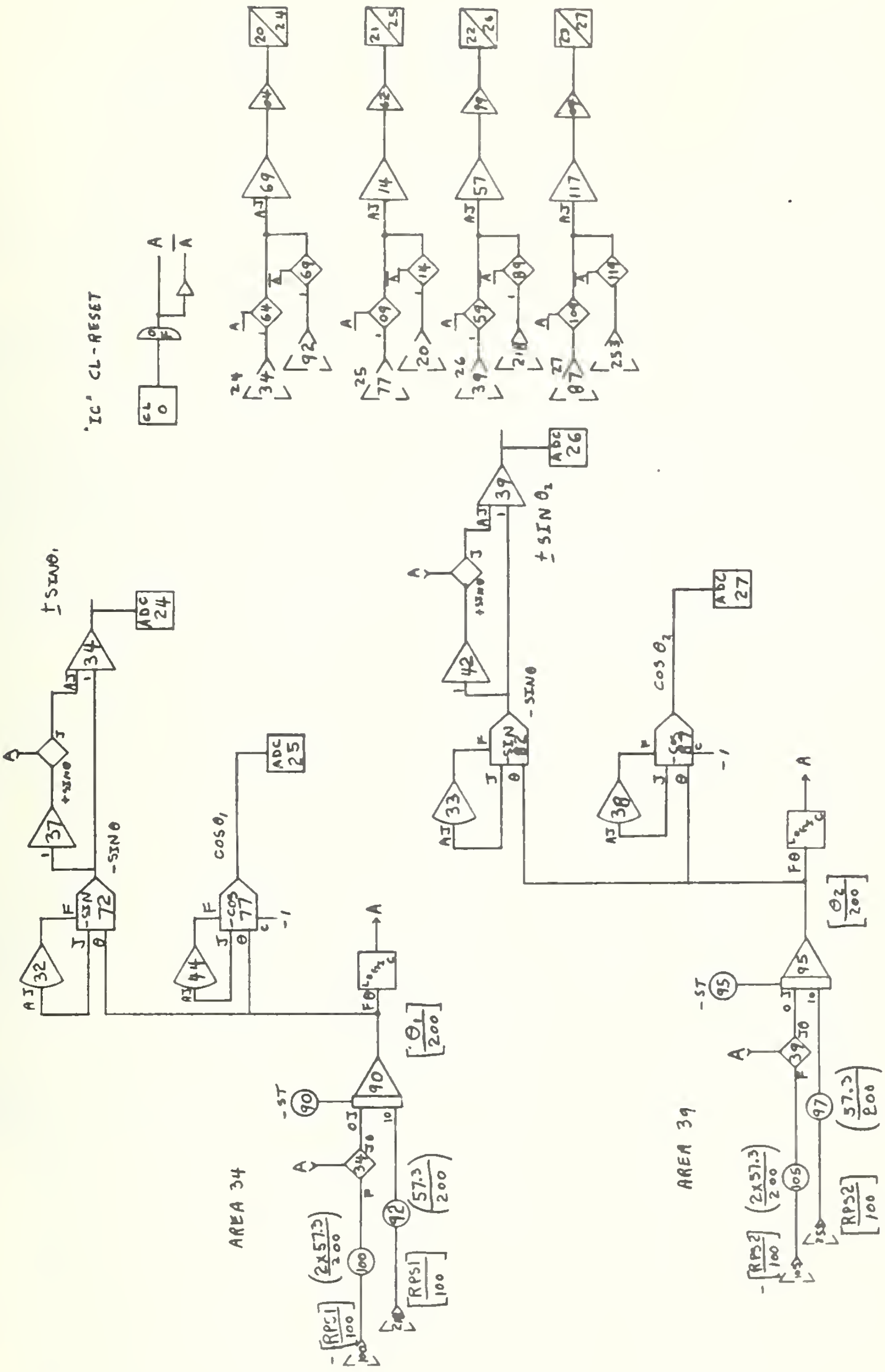


Fig. B-2 ANALOG COMPUTER DIAGRAM – CONTINUOUS RESOLUTION OF WHEEL RATES

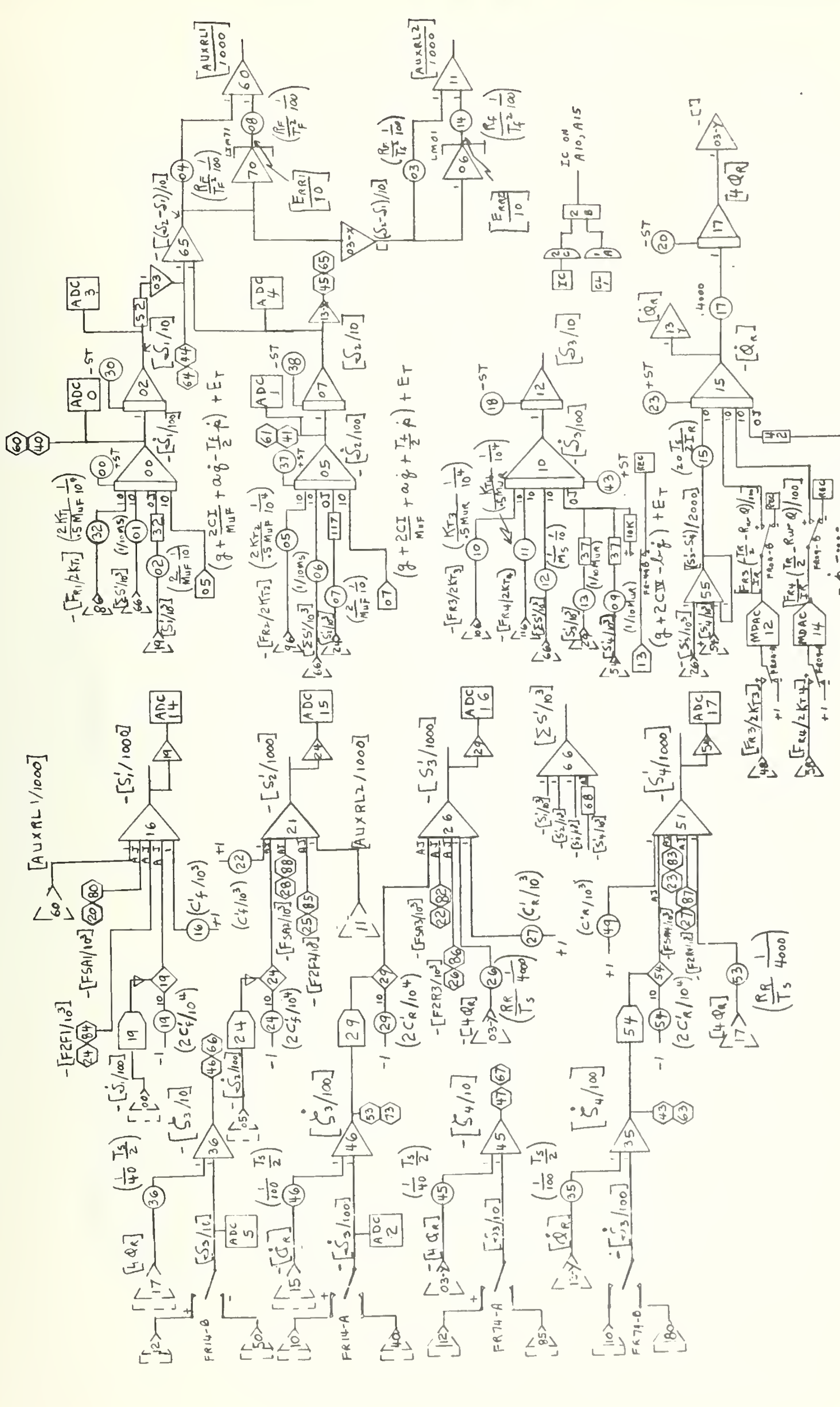
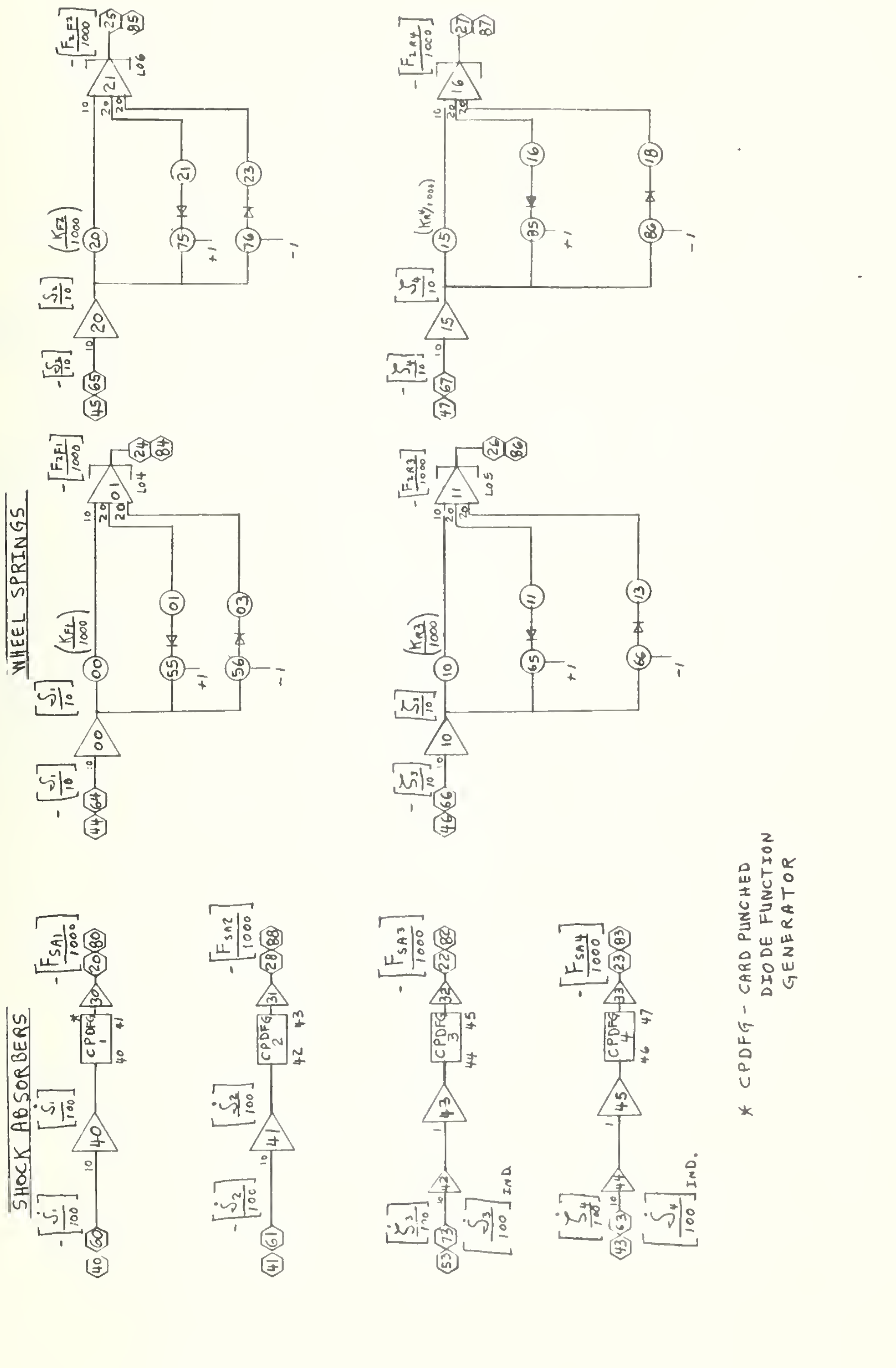


Fig. B-3 ANALOG COMPUTER DIAGRAM – SUSPENSION FORCES AND DEFLECTIONS



* CPDFG - CARD PUNCHED DIODE FUNCTION GENERATOR

Fig. B-4 ANALOG COMPUTER DIAGRAM – WHEEL SPRINGS AND SHOCK ABSORBERS

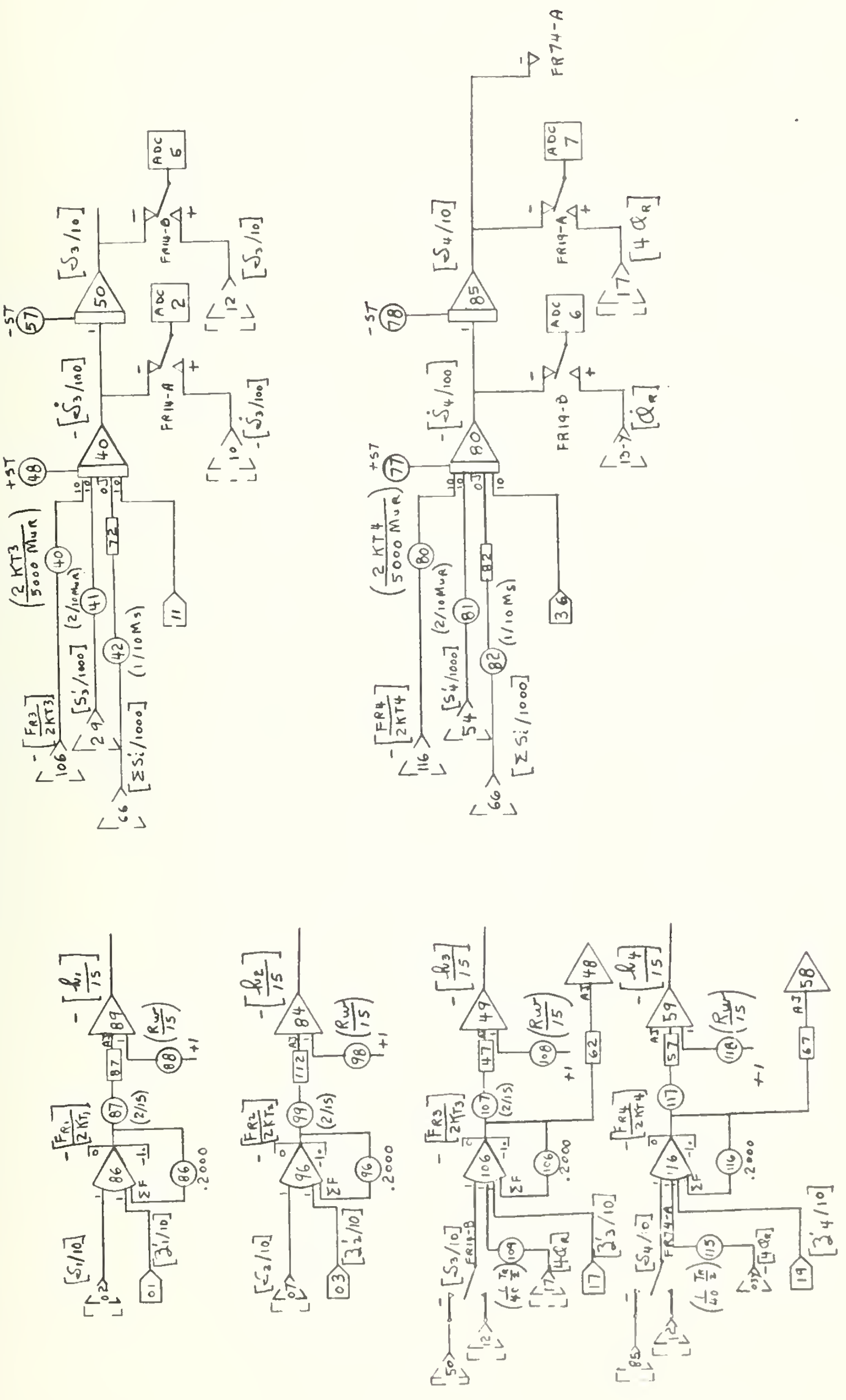


Fig. B-5 ANALOG COMPUTER DIAGRAM – RADIAL FORCES AND INDEPENDENT REAR SUSPENSION

]

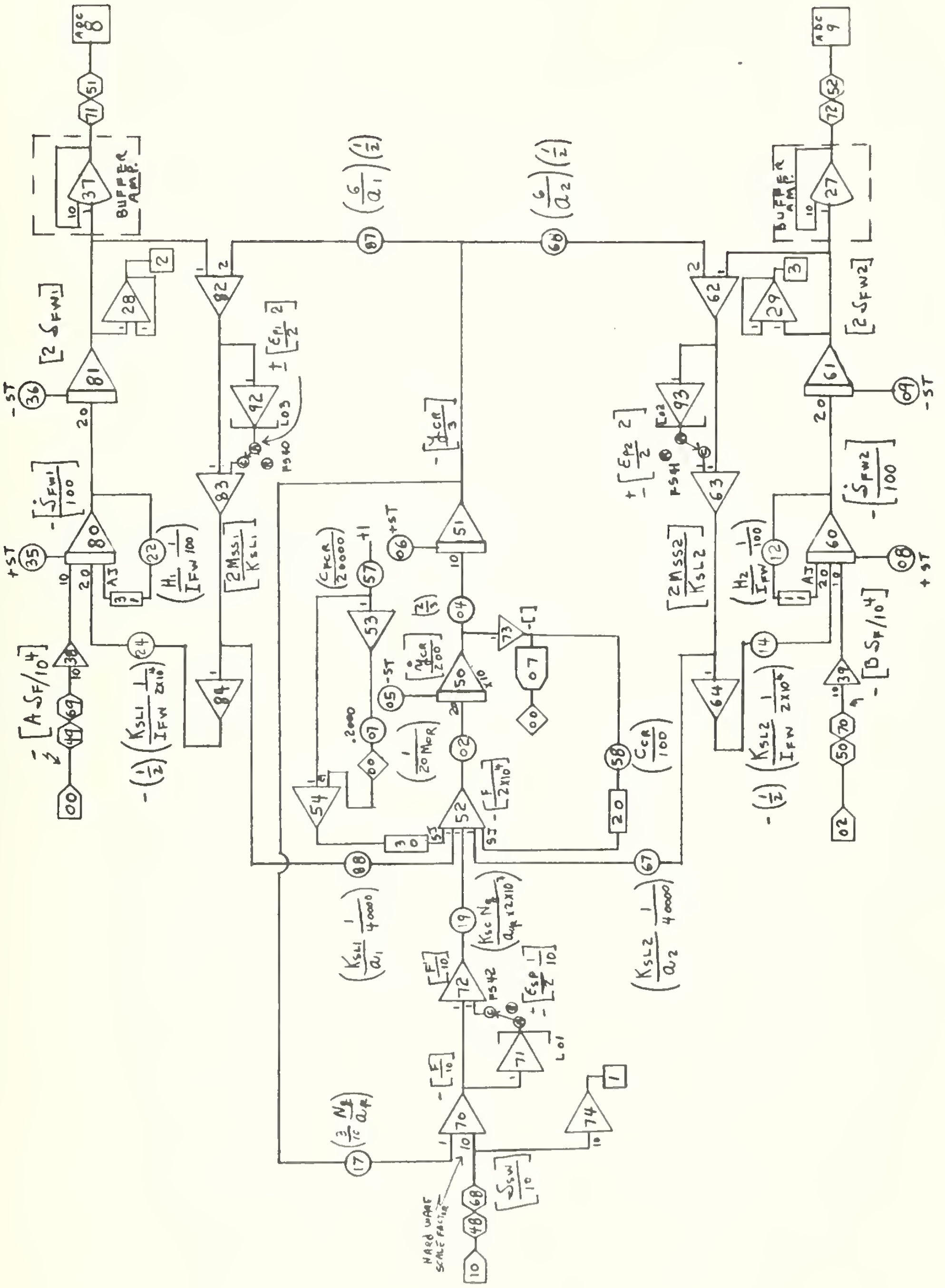


Fig. B-6 ANALOG COMPUTER DIAGRAM – STEERING SYSTEM

4. PRESENTED HERE ARE THE WHEEL SPRINGS AND SHOCK ABSORBER CHARACTERISTICS

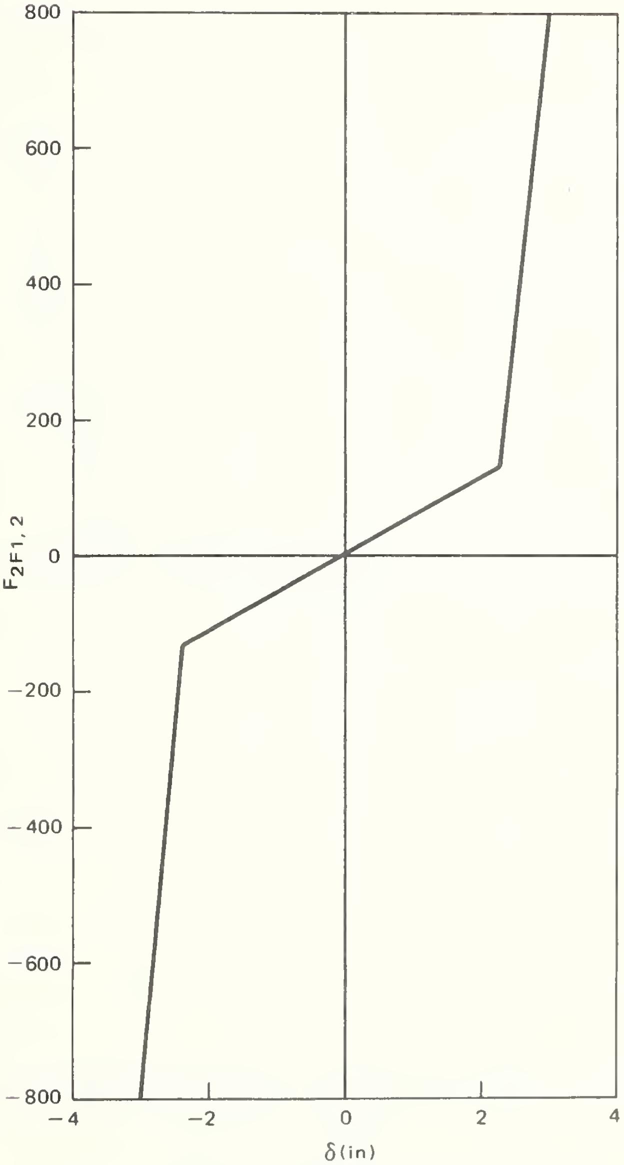


Fig. B-7 FRONT WHEEL SPRING FORCE

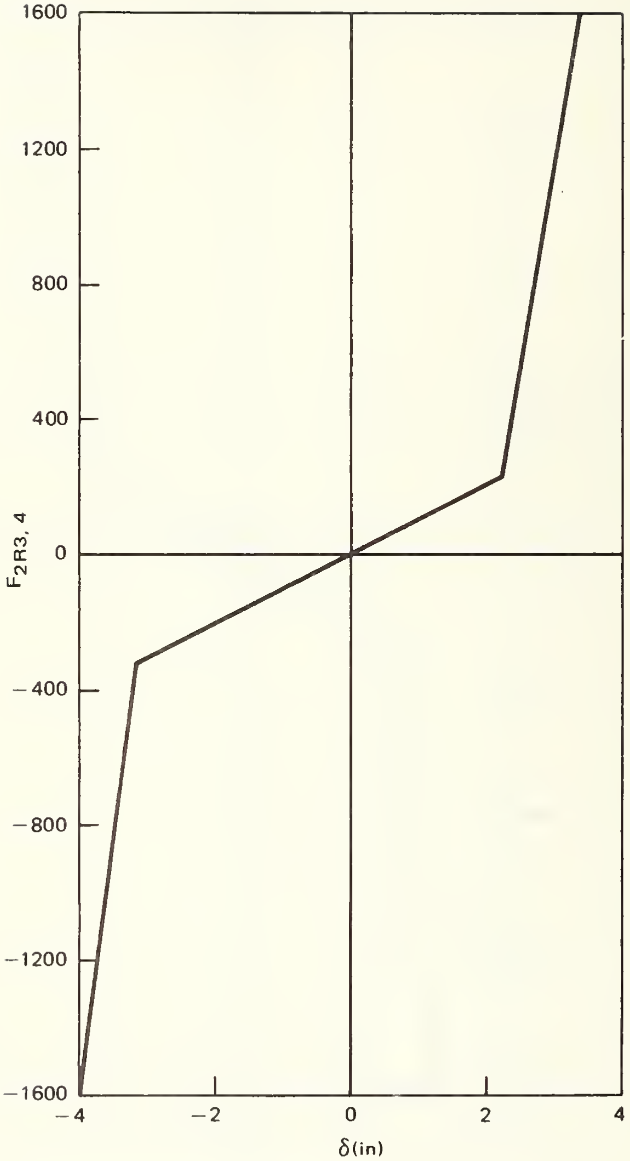


Fig. B-8 REAR WHEEL SPRING FORCE

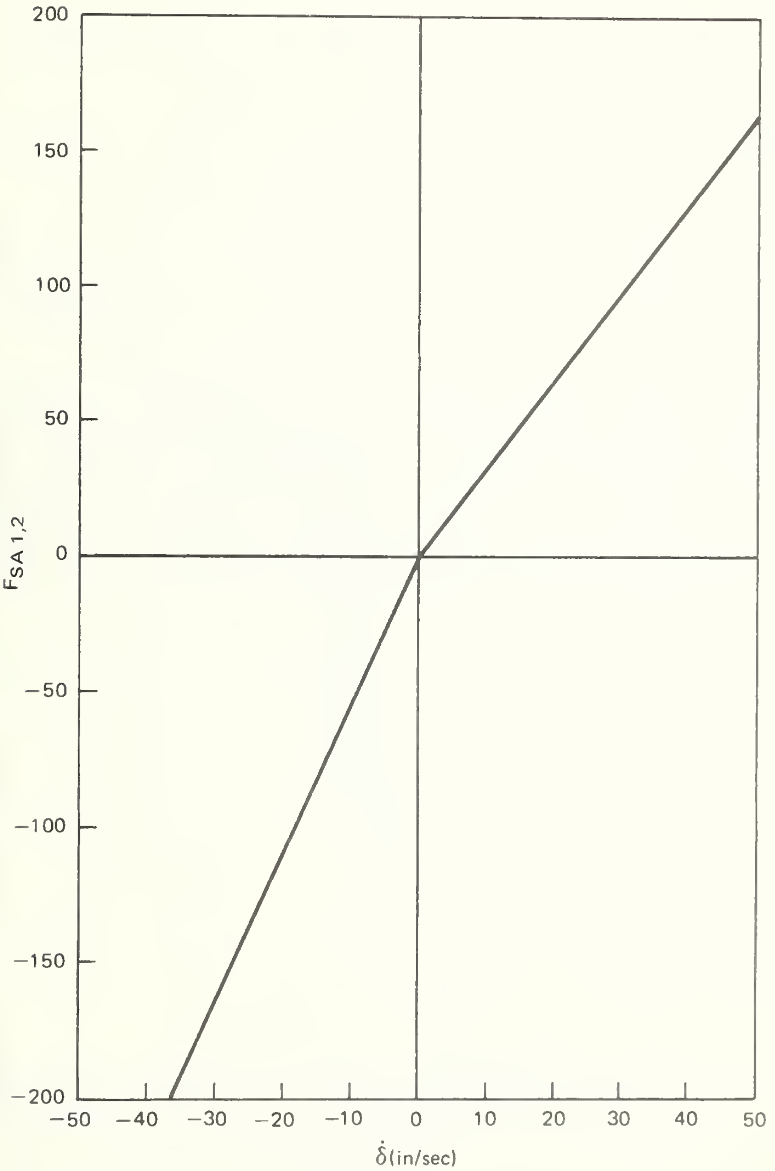


Fig. B-9 FRONT SHOCK ABSORBER VISCOUS DAMPING FORCE

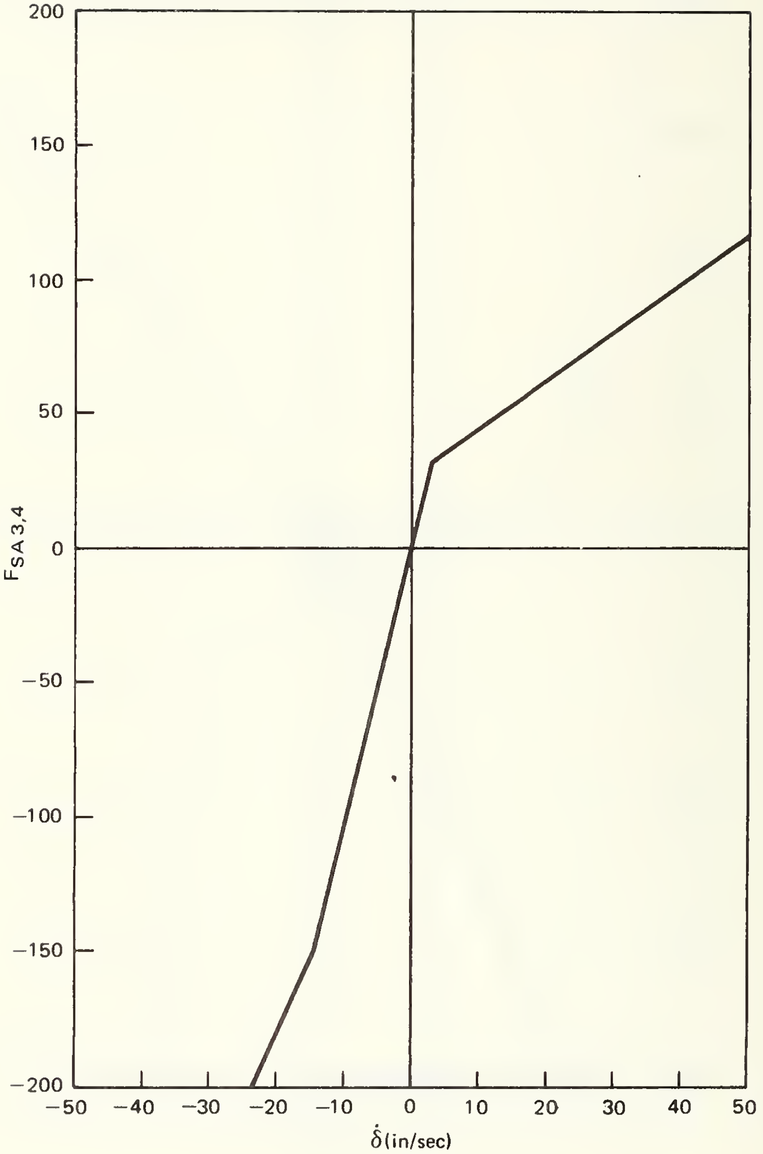


Fig. B-10 REAR SHOCK ABSORBER VISCOUS DAMPING FORCE

5. PRESENTED HERE ARE THE SYMBOLS AND DEFINITIONS OF THE PROGRAM PARAMETERS. THE ORDER OF THE PARAMETERS CORRESPONDS TO THE INPUT DATA CARDS.

SYMBOLS AND DEFINITIONS OF THE PROGRAM PARAMETERS

Card/ Parameter Code	Symbol		Definition or Function (Units)
	Program	Model	
001	MS	M_S	Sprung Mass (lb/in/sec^2)
002	MUF	M_{UF}	Front unsprung mass (lb/in/sec^2)
003	MUR	M_{UR}	Rear unsprung mass (lb/in/sec^2)
004	ZF	Z_F	Distance from c.g. of sprung mass and spin axis of front wheels in z-direction (in)
005	ZR	Z_R	Distance from c.g. of sprung mass and spin axis of rear wheels in z-direction (in)
006	A	a	Distance from c.g. of sprung mass and spin axis of front wheels in x-direction (in)
007	B	b	Distance from spin axis of rear wheels and c.g. of sprung mass in x-direction (in)
008	TF	T_F	Front tread width (in)
009	TR	T_R	Rear tread width (in)
010	TS	T_S	Distance between spring mounts of rear axle in y-direction (in)
011	IX	I_X	Roll moment of inertia of sprung mass (lb-in-sec^2)
012	IY	I_Y	Pitch moment of inertia of sprung mass (lb-in-sec^2)
013	IZ	I_Z	Yaw moment of inertia of sprung mass (lb-in-sec^2)
014	IXZ	I_{XZ}	Product of inertia of sprung mass (lb-in-sec^2)
015	IR	I_R	Roll moment of inertia of rear unsprung mass (lb-in-sec^2)
016	CF		Obsolete: see Card Codes 242-273

Symbols and Definitions of the Program Parameters (cont'd.)

Card/ Parameter Code	Symbol		Definition or Function (Units)
	Program	Model	
017	RF	R_F	Auxiliary roll stiffness in front suspension (in-lb/radian)
018	CFPR	C'_F	Coulomb damping at each front wheel (lb)
019	KF		Obsolete: see Card Codes 234-235
020	LAMF	λ_F	Front spring rate factor at bump stops
021	OMFC	Ω_{FC}	Distance from front suspension equilibrium point to compression bump stop (in)
022	OMFT	Ω_{FT}	Distance from front suspension equilibrium point to tension bump stop (in)
023	CR	C_R	Obsolete: see Card Codes 242-273
024	RR	R_R	Auxiliary roll stiffness in rear suspension (in-lb/radian)
025	CRPR	C'_R	Coulomb damping at each rear wheel (lb)
026	KR	K_R	Obsolete: see Card Codes 236-237
027	LAMR	λ_R	Rear spring rate factor at bump stops
028	OMRC	Ω_{RC}	Distance from rear suspension equilibrium point to compression bump stop (in)
029	OMRT	Ω_{RT}	Distance from rear suspension equilibrium point to tension bump stop (in)
030	KRS	K_{RS}	Rear roll steer (deg/deg)
031	RW	R_w	Undeflected wheel radius (in)
032	B4		Obsolete

Symbols and Definitions of the Program Parameters (cont'd.)

Card/ Parameter Code	Symbol		Definition or Function (Units)
	Program	Model	
033	OMT	Ω_T	Multiple of maximum tire stiffness where cornering stiffness is constant, front wheels
034	A0	A_0	Constant coefficient in cornering stiffness function, front wheels
035	A1	A_1	Linear coefficient in tire cornering stiffness function, front wheels
036	A2	A_2	Quadratic coefficient in tire cornering stiffness function, front wheels
037	A3	A_3	Linear coefficient in tire camber stiffness function, front wheels
038	A4	A_4	Quadratic coefficient in tire camber stiffness function, front wheels
039	ISW	I_{SW}	Moment of inertia of steering wheel (in-lb-sec^2), not operational
040	ANL2	N_{L2}	Steering linkage ratio, lower limit, see Card Codes 134, 135, 138
041	KSC	K_{SC}	Steering column flexibility (in-lb/radian)
042	NG	N_G	Steering gear ratio
043	WG	ω_G	Natural frequency of power steering system (radian/sec), not operational
044	RUN		Run number
045	KSL	K_{SL}	Obsolete, see Card Code 132-133

Symbols and Definitions of the Program Parameters (cont'd.)

Card/ Parameter Code	Symbol		Definition or Function (Units)
	Program	Model	
046	ANL1	N_{L1}	Steering linkage ratio, upper limit, see Card Codes 134, 135, 138
047	IFW	I_{FW}	Moment of inertia of front wheel about its z-axis (in-lbs-sec^2)
048	HDEL	H_δ	Obsolete, see Card Codes 231-232
049	IWF	I_{WF}	Moment of inertia of front wheel about its spin axis (in-lb-sec^2)
050	IWR	I_{WR}	Moment of inertia of rear wheel about its spin axis (in-lb-sec^2)
051	ID	I_D	Moment of inertia of drive line (in-lb-sec^2)
052	AR	$\overline{AR}$	Axle ratio
053	EPS1	ϵ_1	Right front wheel toe-in (radian)
054	EPS2	ϵ_2	Left front wheel toe-in (radian)
055	PT	$\overline{PT}$	Static pneumatic trail or caster offset (in)
056	YSA1	Y_{SA1}	Distance from kingpin axis to wheel center-line, right front (in)
057	YSA2	Y_{SA2}	Distance from kingpin axis to wheel center-line, left front (in)
058	PHS1	φ_{S1}	Kingpin inclination angle, right front (radian)
059	PHS2	φ_{S2}	Kingpin inclination angle, left front (radian)
060	KD	K_D	Drive torque gain factor (in-lb/in/sec)

Symbols and Definitions of the Program Parameters (cont'd.)

Card/ Parameter Code	Symbol		Definition or Function (Units)
	Program	Model	
061	TQD	$\overline{T}_{QD}$	Drive torque (in-lb)
062	K		Not operational
063-074			Initial conditions: $p, q, r, u, v, w, x, y, z, \theta, \varphi, \Psi$. Note that z_0 and θ_0 are computed values and need not be specified.
075	DT	Δt	Integration step size (sec)
076	TN	t_{END}	Maximum run time (sec)
077-078	KTI	K_{Ti}	Tire spring rate, front wheels (lb/in)
079-080	KTJ	K_{Ti}	Tire spring rate, rear wheels (lb/in)
081-084	RPSI	RPS_i	Initial wheel rotation rates computed at t=0 (rad/sec)
085	B1	B_1	Load term coefficient of lateral friction coefficient (1/lb)
086	B2	B_2	Velocity term coefficient of lateral friction coefficient (1/mph)
087	B3	B_3	Constant term of lateral friction coefficient
088	ITSC		
089-106			Initial conditions: $\dot{\delta}_i, \delta_i, \dot{\varphi}_R, \varphi_R, \delta_{FWi}, \mu_{Xi}, S'_i$
107	PPRT		Print control
108	IPRT		Print control
109	ISTP		Run enable switch: 109 = 1 ⟷ no run, 109=0 ⟷ run
110	TQMX		Maximum available drive torque (in-lb)

Symbols and Definitions of the Program Parameters (cont'd.)

Card/ Parameter Code	Symbol		Definition or Function (Units)
	Program	Model	
111	KTQ		
112	VC	V_C	Commanded velocity (mph)
113	MTSW		Multiplier on front wheel aligning torque, M_{Ti}
114	DSWM	δ_{SW}	Maximum steering wheel angle (degrees)
115	TST		Initial time of steer (sec)
116	DSLP		Time to achieve maximum steer angle, equivalent to steer rate, exclude zero (sec)
117	CGAM		Initial time of brake application (sec)
118	CS		Initial time of brake application (sec)
119	TQR		
120	TQF		
121	PFL	P_B	Applied brake pressure (psi)
122	T1		
123	DSW		Sinusoidal steer amplitude (degrees)
124	TSW		Duration of sinusoidal steer (sec)
125	ISW5		VHTP sinusoidal steer enable code: 125=1 ↔ enable
126	SW15		VHTP roll over enable code: 126=1 ↔ enable
127			Equation suppress option, 127=0 ↔ none, 1 ↔ $\dot{p}=0$, 2 ↔ $\dot{q}=0$, 3 ↔ $\dot{p}=\dot{q}=0$

Symbols and Definitions of the Program Parameters (cont'd.)

Card/ Parameter Code	Symbol		Definition or Function (Units)
	Program	Model	
128			VHTP index
129			Obsolete
130	AMCR	M_{CR}	Mass of steering system connecting rod (lb-sec^2/in)
131	ESP	ϵ_{SP}	Steering wheel deadband (deg)
132	KSL1	K_{SL1}	Steering linkage flexibility, right front wheel (in-lbs/radian)
133	KSL2	K_{SL2}	Steering linkage flexibility, left front wheel (in-lb/radian)
134-135	AAI	a_i	Length of steering linkage arms (in). Note $N_L = a_L/a_P$.
136	CCR	C_{CR}	Viscous damping coefficient, steering system (lb/in/sec)
137	CFCR	C_{FCR}	Coulomb damping, steering system (lb)
138	AP	a_P	Length of Pitman arm (in). Note $N_L = a_L/a_P$.
139-140	EPI	ϵ_{Pi}	Steering linkage play (degrees)
141-142	ERRI	ϵ_{Ri}	Auxiliary roll stiffness play
143-144	AMLI	m_{li}	Unbalanced wheel mass (lb-sec^2/in)
145	RRIM	R_{rim}	Wheel rim radius (in)
146	RWR	R_{WR}	Wheel rim width (in)
147-169			Not operational, ABS parameters
170	SN10		Road surface skid number
171	SN11		Road surface skid number

Symbols and Definitions of the Program Parameters (cont'd.)

Card/ Parameter Code	Symbol		Definition or Function (Units)
	Program	Model	
172-187			Not operational, ABS parameters
188	TBSW		Summary print control, 188=0 ↔ suppresses print, 188=1 ↔ print stored table
189			Table store control, 189=0 ↔ no table storage, 189=1 ↔ store summary table
190-195			Not operational, ABS parameters
196-197	EKI	ϵ_{Ki}	Static front wheel toe-in offset angles (degrees)
198	BMPL	W_B	Length of single road bump (in)
199	BMPS	L_B	Distance between road bumps (in)
200	BMPH	h_B	Road bump height (in)
201	XB	X_B	Initial distance from car to first bump (in)
202-218			Special function coefficient entries, not operational
219-220	FEEI	F_{EEi}	Front wheel camber offset angles (degrees)
221-222	THEI	θ_i	Front wheel caster offset angles (degrees)
223-230	CRIC(T)		Shock absorber damping derivatives, compression and tension, by wheel, not operational
231-232	HI	H_i	Viscous damping in front wheel azimuth dynamics (lb/in/sec)
233	NI		Obsolete
234-235	AKFI	K_i	Front suspension spring rates (lb/in)

Symbols and Definitions of the Program Parameters (cont'd.)

Card/ Parameter Code	Symbol		Definition or Function (Units)
	Program	Model	
236-237	AKFJ	K_i	Rear suspension spring rates (lb/in)
238-241	BRI	λ_{Bi}	Brake imbalance factors
242-257	SSJI	F_{SAji}	Shock absorber damping force values by segment and wheel j-segment index, i-wheel index, segments taken from $-\dot{\delta}$ to $+\dot{\delta}$ (lb/in/sec)
258-265	BPJI	$\dot{\delta}_{(j)}$	Non-zero break points of segmented shock absorber characteristic (in/sec)
266-273	LBJI	$F_{SA(j)}$	Shock absorber damping force at break points (lb)
274	TLF	T_{LF}	Not operational, anti-roll suspension lever arm (in)
275	ZRO	Z_{RO}	Not operational, anti-roll suspension lever arm (in)
276	LF	L_F	Not operational, anti-pitch suspension lever arm (in)
277	BMPN	N_B	Number of bumps in bump grid
278	TQB0	t_{B0}	Time of brake application in combined severe brake and steer VHTP
279	TQB1	t_{B1}	Time of brake release in combined severe brake and steer VHTP
280			Enable code to input an entire new parameter set, enable value=1
281	TSF	T_{SF}	Multiplier on front wheel tire forces
282	TSR	T_{SR}	Multiplier on rear wheel tire forces
283	TANP	tanp	Anti-pitch factor in rear suspension
284	HFC	h_{FC}	Roll center of front suspension (in)

Symbols and Definitions of the Program Parameters (cont'd.)

Card/ Parameter Code	Symbol		Definition or Function (Units)
	Program	Model	
285	HRC	h_{RC}	Roll center of rear suspension (in)
286	CRRC	C_3	Linear coefficient in solid rear axle roll center dynamics
287	AXLE		Solid rear axle/split rear axle option code, solid=1, split=2
288-289	PSRI	Ψ_{Ri}	Not operational, split rear axle trail angles (degrees)
290	ROT	Ω_T	Multiple of maximum tire load where cornering stiffness is constant, rear wheels
291	RA0	RA_0	Constant coefficient in tire cornering stiffness, rear wheels
292	RA1	RA_1	Linear coefficient in tire cornering stiffness function, rear wheels
293	RA2	RA_2	Quadratic coefficient in tire cornering stiffness function, rear wheels
294	RA3	RA_3	Linear coefficient in tire camber stiffness function, rear wheels
295	RA4	RA_4	Quadratic coefficient in tire camber stiffness function, rear wheels

APPENDIX C

DESCRIPTION OF HYBRID COMPUTER SIMULATION LABORATORY

1. HYBRID COMPUTER

The APL hybrid computer is unique in that the analog computer is a terminal to a large digital computer, with communication between them occurring over a 1000-foot data path. A block diagram of the system is presented in Figure C-1. The analog computer is an EAI 680 and the digital computer is an IBM System 360 Model 91. The data channel that provides the high-speed transmission required for hybrid processing is the parallel data adapter (PDA) of the IBM 2909 asynchronous data channel. The 2909, which is capable of transmitting at a 2M-byte rate, is working at about 40% of this rate (200K 16-bit words per second) through the PDA while servicing the hybrid. Besides the PDA, the 2909 has two additional subchannels used for hybrid processing.

The priority interrupt subchannel provides up to 64 priority interrupt levels (PILS) for a System 360 which, with the exception of the 360/44, does not have a priority interrupt structure. Thus, this subchannel was necessary in order to provide the hybrid system with PILS. This subchannel currently is expanded to 16 PILS. The other subchannel provides a precision interval timer (1 μs resolution) to the hybrid user. This timer can be loaded via software and causes an interrupt when decremented to zero, which in turn initiates an event in a hybrid program.

The 360/91, one of the largest and fastest computers built by IBM, has the following characteristics:

- Third-generation hardware
- 4 million bytes of main core storage
- 1.5 billion bytes of direct access storage
- Minimum execution time of 60 ns per instruction
- Uses the Operating System OS/MVT (Multiprogramming with a Variable Number of Tasks)

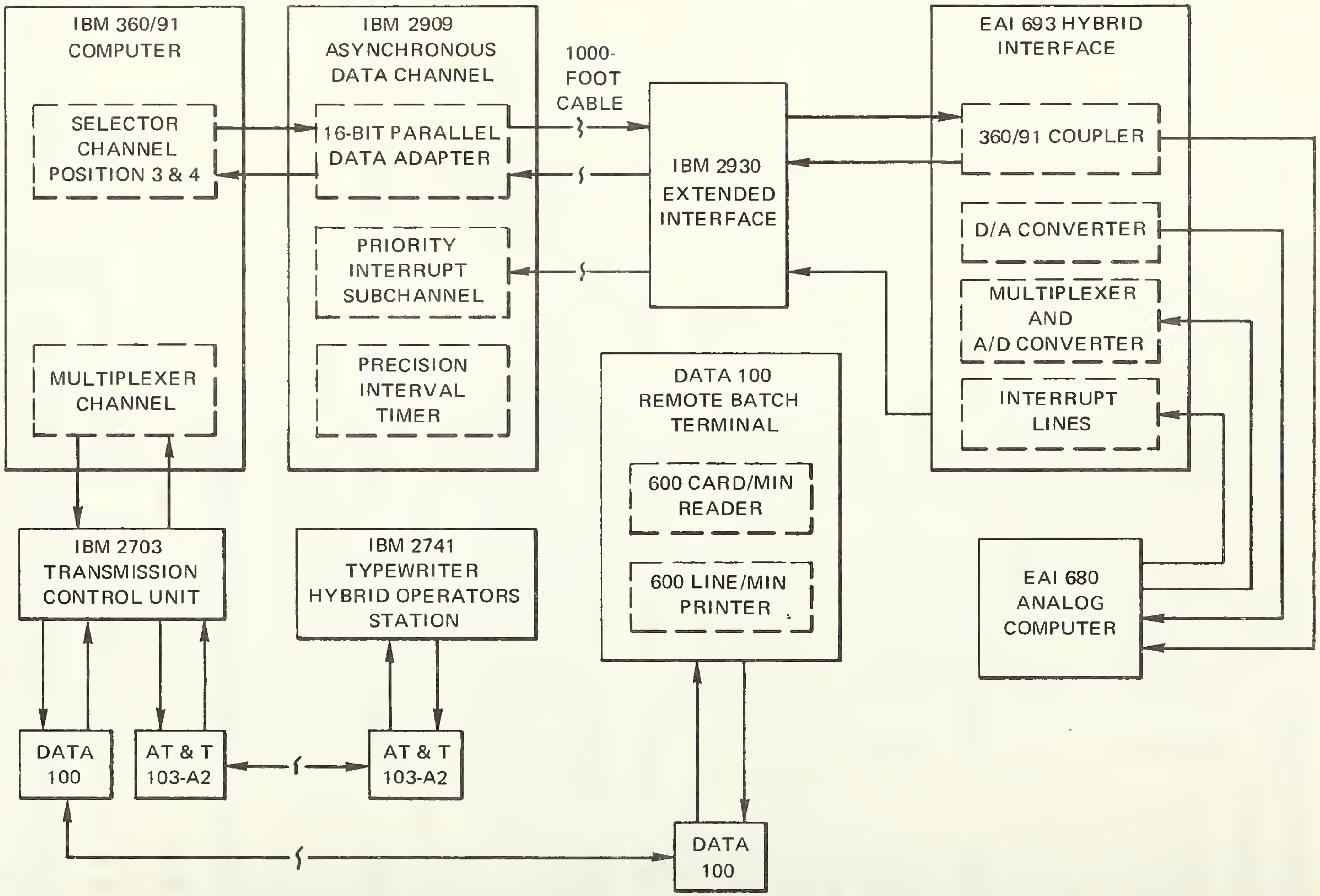

IBM 360/91 COMPUTER
SELECTOR CHANNEL POSITION 3 & 4
MULTIPLEXER CHANNEL
IBM 2909 ASYNCHRONOUS DATA CHANNEL
16-BIT PARALLEL DATA ADAPTER
PRIORITY INTERRUPT SUBCHANNEL
PRECISION INTERVAL TIMER
1000-FOOT CABLE
IBM 2930 EXTENDED INTERFACE
EAI 693 HYBRID INTERFACE
360/91 COUPLER
D/A CONVERTER
MULTIPLEXER AND A/D CONVERTER
INTERRUPT LINES
EAI 680 ANALOG COMPUTER
DATA 100 REMOTE BATCH TERMINAL
600 CARD/MIN READER
600 LINE/MIN PRINTER
IBM 2703 TRANSMISSION CONTROL UNIT
IBM 2741 TYPEWRITER HYBRID OPERATORS STATION
DATA 100
AT & T 103-A2
AT & T 103-A2
DATA 100

Fig. C-1 HYBRID SYSTEM BLOCK DIAGRAM

Two advantages are immediately obvious to a 360/91 interactive user. First, since he is only charged in proportion to his use of the central processing unit (CPU) and main core storage in the multi-programming environment, the machine is cheap. Second, since the 360/91 is a third-generation computer with nearly unlimited resources, it is extremely fast and efficient when computing. In summary, this computer gives the interactive terminal user unlimited digital resources at a cost comparable to that of a much smaller computer.

The Interactive Simulation Laboratory (ISL) is located 800 feet from the 360/91 digital computer. The 2909 PDA, which has specially designed line drivers and receivers (now standard from IBM), can easily drive the 1000-foot cable to the ISL. However, matching drivers and receivers are required in the ISL. These are housed in the IBM 2930. Individual lines are required for each bit of the data word, each control line, and for each PIL. The 2930 then cables into a 360/91 coupler (standard EAI product) housed within the EAI 693. From the coupler on, the 693 hybrid interface and 680 analog computer are standard EAI equipments.

In addition to the hybrid hardware of Figure C-1, the ISL contains two pieces of equipment that support the hybrid operations. These are a Data 100 remote batch terminal and an IBM 2741 typewriter. The Data 100 consists of a 600 card/minute reader and a 600 line/minute printer which allow the submission of jobs to the 360/91 and the return of results from the 360/91 at the ISL hybrid work station. The Data 100 operates over a dedicated telephone line into a controller connected to the 360/91 multiplexer channel. Its support package is the IBM remote job processing (RJP) portion of the ASP job scheduler. The 2741 has evolved as the hybrid control console. Via APL-written software, all hybrid jobs can be interfaced to the 2741 and problem execution directed from it. The 2741 operates over a switched line into the multiplexer channel via a controller.

These hardware items comprise the APL hybrid computer system. In addition to this system, the ISL contains three 231-R analog computers, each interfaced to an EAI DES-30 logic unit and an EAI quarter-square multiplier rack. These machines are interfaced to the hybrid system via buffering amplifiers for large simulations.

2. HYBRID SOFTWARE

Included with the original EAI hardware purchase were three hybrid software packages operable under Operating System 360 (OS/360). The first, the HYIOS (Hybrid Input/Output Subroutines) package, was a collection of Fortran IV callable subroutines that allowed operation from the 360 of all hybrid functions in the 680 and 693 originally designed for control from the EAI 640 digital computer (Ref. 3). These routines number 47 and permit complete control and readout of the data interface, logic components, and analog components. Also, these routines do extensive error checking and each has an error argument returned, which can be checked by the programmer to determine successful completion of a hybrid operation. The routines and the hardware have proven so reliable that these arguments are seldom checked in current hybrid programs.

The second software package delivered by EAI was HYSAT (Hybrid Static Analog Test). HYSAT is an interpretive language written in Fortran using HYIOS, which automates the performance of analog computer setup and static checks. Simple commands allow changing the analog mode, setting potentiometers and control lines, reading the outputs of amplifiers, integrator derivatives, and track-store inputs, and reading the state of sense lines, comparators, and relays. In the verify mode, the outputs of the analog and logic components can be compared against predetermined values, with all errors documented on the 2741. This program is interactive from the 2741 or from cards. The capabilities of HYSAT have been combined with ease of programming of a simulation language (DSL/91) to produce a completely automated static check procedure for hybrid programs (Ref. 4). A specially wired board and specially prepared HYSAT data deck are used to verify that all 680 analog components are operational.

Whereas HYSAT permits checking the 680 analog console statically, the third software package delivered by EAI permits checking the 693 hybrid interface. HIORT (Hybrid Interface Operational Readiness Test) is again an interpretive language that is interactive from the 2741 and is used to perform A/D converter absolute error tests and A/D - D/A converter relative error tests. With this program, information is obtained to demonstrate the functioning, within specification, of the hybrid interface.

Thus, with these three software packages, complete hybrid programming is obtained. HYIOS provides the Fortran callable routines for application programs; HYSAT provides a convenient interactive method for performing initial setup and static checks; and HIORT provides an interactive method of insuring the accuracy of the hybrid interface. These programs are fully documented in Ref. 3.

3. REAL-TIME HYBRID

The multiprogramming 360/91 is a great asset for the hybrid in that it provides nearly unlimited digital resources and inexpensive operation. However, when real-time hybrid is considered, multiprogramming becomes a liability. Since many users are time-sharing the facilities of the large central processor, service within a fixed time frame cannot be guaranteed. From the beginning, it was obvious that a hypervisor was required for real-time jobs.

This need has been met by IBM's real time monitor (RTM). RTM was developed at the IBM Palo Alto DACS Development Center and was first reported in a seminar at the DACS Center in September 1968 (Ref. 5). As first reported, RTM guarantees CPU service to jobs designated as real-time jobs, with RTM co-resident with the OS/360 monitor. The real-time job can use all the facilities of OS and can force all non-real time jobs into the "wait state" when it needs the CPU. The original hardware involved in the RTM development was an IBM 360/50 working through a "transparent" IBM 1800 into an Applied Dynamics (AD-4) analog computer. In mid 1969, the DACS Development Center agreed to augment their basic RTM with the code required, to have RTM support the PDA, precision interval timer, and priority interrupt subchannels of the IBM 2909. These packages provide Fortran callable subroutines as well as assembly language instructions for accessing and controlling the three subchannels. RTM is a "free" software package for those who use OS/360 operating systems; it was released in December 1970. Reference 6 is an introduction to RTM and Ref. 7 is the RTM Operations and Programmer's Guide.

Several applications programs have been written using the real-time monitor; however, two basic types of operation are employed: (a) initialization of real-time events via the priority interrupts and (b) the initialization of real-time events via decrementing of the interval timer. Besides the conventional closed-loop hybrid technique of executing a digital program involving A/D conversion, digital calculation, and D/A conversion, an interrupt-based, multiplying D/A converter, hybrid function generation scheme (Ref. 8) has also been implemented. In order to complete support of these real-time hybrid jobs, the HYIOS subroutines were rewritten to run under RTM as well as OS/360. During this rewrite much of the error checking in these routines was omitted, greatly decreasing the execution time of each routine.

4. HYBRID OPERATION

Two modes of hybrid operation are currently employed: closed-loop real-time hybrid and data acquisition. In both cases the hybrid jobs are submitted to the 360/91 via the Data 100. If it is a real-time job, the hybrid applications engineer must have RTM released prior to the release of his application program. Both real-time and data acquisition jobs are run interactively from the 2741 typewriter and jobs are available within minutes of submission.

Hybrid data acquisition programs are those that use the digital computer to vary parameters interactively between runs in an all-analog simulation, collect data at the end of a run, and then post-process the data to determine a cost function. No real-time constraint exists in this type of running, and stochastic variation (white noise) is usually an input. Many runs are required to statistically define a data point. The digital computer response is gotten by forcing the 360/91 program into the "wait state" for the analog solution time immediately following the instruction to change the analog mode to operate. The statements following the "wait" collect the data and then return control to the typewriter. This mode has been quite reliable and control is immediately returned to the program following the "wait". Non-real time jobs operate under OS/360 at a software priority of 11 out of 13, using the HYIOS routines supplied by EAI. As previously mentioned, all real-time jobs operate under RTM in the manner discussed.

REFERENCES

1) Task Z700 under Contract No. N00017-72-C-4401.

2) "Final Report, Vehicle Handling," U.S. D.O.T., N.H.S.B. Contract FH-11-6946, dated April 30, 1970.

3) "EAI 680/IBM System 360 Reference Handbook," Electronic Associates, Inc., Publication No. 00800.3044-1, dated March 1970.

4) P. F. Bohn, "Simulation Language Generated Static Checks for Hybrid and Analog Simulations," Simulation, dated September 1971.

5) J. Clark, "The Real-Time Monitor," presented at a hybrid seminar at DACS Development Center.

6) "Introduction to the Real-Time Monitor (RTM)," IBM Publication No. GH20-0824-0, dated July 1970.

7) "Real Time Monitor (RTM) Operations and Programmer's Guide," IBM Publication No. GH20-0877-0, dated October 1970.

8) A. I. Rubin, "Hybrid Techniques for Generation of Arbitrary Functions," Simulation, dated December 1966.

9) "Computer Simulation of Vehicle Handling," U.S. D.O.T., NHTSA Contract FH-11-7563, dated September 1972.

10) "Automobile Dynamics - A Computer Simulation of Three-Dimensional Motions for Use in Studies of Braking Systems and of the Driving Task," CAL Report No. VJ-2251-V-7, dated August 1970.

78243

TL 243
.B65

BORROW

~~SiD Wi~~

Form DOT F 1720
FORMERLY FORM DO

DOT LIBRARY
00092833